UNIVERSITY OF BAHRAIN

College of Information Technology

Computer Science Department

Machine Learning Techniques for Intrusions Detection Improvements in Supervisory Control and Data Acquisition

A Thesis Submitted in Partial Fulfilment of the Requirements for the **Ph.D.** Degree in

Information Technology

Submitted By

Hanan Fouad Hussain Alwadi
202012498

Supervised By

Dr. Mohsen Rouached
أستاذ مشارك
جامعة البحرين

May 2025

University of Bahrain
Deanship of Graduate Studies
And scientific Research

جامعة البحرين
عمادة الدراسات العليا والبحث العلمي

Dissertation Deposit Form

Author Name:	Hanan Fouad Hussain Alwadi		
Email:	202012498@stu.uob.edu.bh	Contact No:	00966503805039
College:	**Information Technology**		
Department:	**Computer Science**		
Thesis Title:	Machine learning techniques for intrusion detection improvement in supervisory control and data acquisition.		
Year:	2025		

Author's Declaration: I agree to the following conditions:

The above thesis will be available (in the University of Bahrain library and externally) and reproduced as necessary at the discretion of the University of Bahrain. It may also be digitized by the University of Bahrain and made available via the university's repository or via other parties on the Internet.

This condition shall apply to ALL copies including electronic copies.

The above thesis has been provided on the understanding that it is copyright protected material and that no quotation from this thesis may be published without proper acknowledgement.

Signature of the Author:	Hanan	Date:	22\10\2025

Approved at the University Council 6th meeting, 2013
In accordance with Decision No. 736/2013 dated 14/3/2013
Modified by University Council Decision No. 2043/2013 dated 20/11/2013.

DECLARATION

I declare that this work is the result of my own investigation and that it has not already been accepted in substance for any degree, nor is it currently submitted for any degree.

Signed: ---------- Date: 7/10/2025

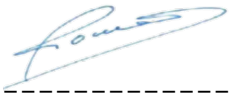
Hanan Fouad Hussain Alwadi (Candidate)

The Thesis defense committee considers the thesis titled:

Machine Learning Techniques for Intrusions Detection Improvements n Supervisory Control and Data Acquisition

Submitted by Hanan Fouad Hussain Alwadi to the College of Information Technology is satisfactory and acceptable for the PhD degree in Computing and Information Sciences.

The Defense Committee:



Dr. **Mohsen Rouached** Department of Computer Science
(Supervisor) University of Bahrain

Kevin Curran

Prof. **Kevin Curran** Ulster University
(Member) UK



Dr. **Wael Madany** Department of Computer Engineering
(Member) University of Bahrain

Abstract

Supervisor Control and Data Acquisition (SCADA) networks are extremely important in the control of infrastructure considered vital to the digital age including power grid, water supply systems, and industrial facilities. The issue of resilience of these networks to changes in cyber threats has become a key focus in research. This paper explores using various machine learning (ML) models to improve intrusion detection and adversarial resistance in SCADA operating conditions using the suggested SCADA Protection Hub Framework. According to the Quantitative Results, the framework consists of five fundamental models, such as Decision Tree, Support Vector machine (SVM), Random Forest, Neural Network and a Voting Classifier Ensemble to enhance accuracy and reliability of detection. There were significant experiments done on benchmark SCADA intrusion datasets that the ensemble model obtained an overall accuracy of 98.7, precision of 98.4, recall of 97.9 and F1-score of 98.1. Comparative confusion matrix analyses were also utilized to investigate the true and false classifications and give an insight about the strengths and weaknesses of each of the models when it comes to identifying covert deception and adversarial attacks. The study also investigated adversarial robustness by simulating adversarial pertussis and assessing the sensitivity of each model to manipulated inputs. The analysis of the time series trends of intrusions also indicated the key time trends that can underpin the proactive threat detection and response to the threat of intrusion. The computational resource requirements, scalability, and practical feasibility of the real-time implementation of SCADA are also discussed in the proposed framework. In general, the results indicate that the SCADA Protection Hub Framework has a significant effect on increasing the reliability of detection, adversarial resilience, and situational awareness in industrial control. The present study can be elaborated in the future by incorporating federated learning alongside edge-based detection tools to enhance distributed security of heterogeneous SCADA infrastructure.

Publications

Hanan Fouad Alwadi, Mohsen Rouached, Feature Selection Techniques and Time Series Predictions: Solutions for Accurate Energy Demand Forecasts. Nanotechnology Perceptions ISSN: 1660-6795, E-ISSN: 2235-2074. July 05, 2024. https://nano-ntp.com/index.php/nano/article/download/1181/1029/2118

Dedication

This thesis is dedicated to the remarkable individuals who have played an integral role in my academic journey and personal growth.

To my family, who has always supported me and made me feel strong: to my parents, who have always believed I could achieve anything I set my heart on. I could not be who I am today without your sacrifices and teaching and I thank you for teaching me to be strong and hardworking.

I want to thank my partner for giving me love, patience and understanding during every difficult moment of this trip. Because you believed in me, I've had the willpower to overcome difficulties and stay focused. I appreciate having you as my confidant, as you have been there for all I've been through along the way.

I am deeply grateful to the advice and support from my mentors and advisors that have helped shape my education. You feedback and understanding have made my research stronger and encouraged me to approach tasks from both sides of my brain. Thank you for all the opportunities to learn and for the wonderful things you have taught me.

I appreciate my friends and colleagues for joining me on this journey and for all the support they've given. Staying up late to study, chattering about solutions and laughing with everyone else have made these moments stand out. Your support and understanding have made dealing with graduate life's ups and downs more comfortable.

This work is truly dedicated to those who have supported and mentored me or simply spent time with me during my journey. The success of this thesis is thanks to the encouragement and support you have given me each day. Thanks for being involved in this experience.

Acknowledgments

I take this opportunity to acknowledge the guidance and support provided by my advisor and mentors, whose valuable insights have helped shape my research.

I also thank my family and friends for their encouragement throughout this journey. This work stands as a testament to my commitment, dedication, and integrity in conducting independent research.

Table of Contents

Abstract III

Chapter 1: Introduction 1

1.1. Background and Context 1

1.2. Problem Statement 3

1.3. Objectives of the Study 5

1.4 Significance of the Research 7

1.5 Overview of Machine Learning in Intrusion Detection 9

1.6 Limitations of the Study 11

1.7 Chapter Organization 13

1.8 Conclusion of chapter 15

Chapter 2: Literature review 16

2.1 Historical Evolution of SCADA Systems 16

2.1.1 Early Origins and Development 16

2.1.2 Transition to Digital SCADA Systems 17

2.2 SCADA Security Concerns 19

2.2.1 Importance of Security in Critical Infrastructure 20

2.2.2 Common Vulnerabilities in SCADA Systems 22

2.2.3 Cyber Attacks on SCADA Networks 24

2.2.4 Modern-day SCADA: Features and Advancements 26

2.3 Intrusion Detection: An Overview 28
2.3.1 Basics of Intrusion Detection Systems (IDS) 28
2.3.2 Signature-based vs. Anomaly-based Detection 30
2.3.3 Challenges in SCADA Intrusion Detection 33
2.4 Machine Learning in Cybersecurity 35
2.4.1 Fundamentals of Machine Learning 35
2.4.2 Application of Machine Learning in Cyber Threat Detection 37
2.4.3 Strengths and Weaknesses of ML-based IDS 38
2.5 Existing Machine Learning Techniques for SCADA Intrusion Detection 42
2.5.1 Supervised Learning Approaches 42
2.5.2 Unsupervised Learning Approaches 43
2.5.3 Hybrid Models and Ensemble Techniques 45
2.6 Evaluation Metrics for IDS 46
2.6.1 Accuracy, Precision, and Recall 46
2.6.2 False Positive and False Negative Rates 49
2.6.3 ROC Curves and AUC Values 51
Chapter 3: Stealthy Deception Attacks in SCADA Networks 54
3.1 Conceptual Framework 54
3.2 Architectural Analysis 57
3.3 Examples and Real-Life Case Studies 61

3.4 Sensitivity Points and Attack Scenarios 65

Chapter 4: Machine Learning in SCADA system 74

4.1 Introduction to Machine Learning in SCADA Security 74

4.2 Model Evaluation and Selection 80

4.3 Adversarial Machine Learning in SCADA Systems 87

4.4 Real-Time Detection Challenges in SCADA Systems 93

4.5 Time Series Analysis for Intrusion Detection in SCADA Systems 101

Chapter 5: Data Acquisition and Modeling Methodology 109

5.1 Introduction 109

5.1.1 Context and Motivation 109

5.1.2 Objectives of the Data Collection and Modeling Framework 110

5.2 Technical Overview of the SCADA Environment 111

5.2.1 SCADA System Architecture and Functional Components 111

5.1.1 Communication Protocols and Data Transmission Pathways 112

5.1.2 Known Vulnerabilities and Potential Attack Surfaces 112

5.3 Data Acquisition and Preprocessing Pipeline 113

5.3.1 SCADA Data Sources and Collection Techniques 113

5.3.2 Data Cleaning, Normalization, and Quality Checks 115

5.3.3 Feature Engineering: Extraction and Selection 116

5.4 ML Model Design and Selection 116

5.4.1 Model Selection Criteria Based on SCADA Requirements....117
5.4.2 Technical Overview of Selected Machine Learning Models....118
5.5 Implementation of Stealthy Deception Attack Detection.... 122
5.5.1 Security Challenges in SCADA and SDA Threat Landscape.... 122
5.5.2 Modeling and Understanding Stealthy Deception Attacks.... 123
5.5.3 Integration of ML Models with the Attack Framework.... 124
5.5.4 Rule Extraction and Hybrid Integration with Signature-Based IDS.... 125
5.5.5 Deployment of the Proposed Intrusion Detection System.... 126
5.6 Model Training, Validation, and Performance Measurement.... 132
5.6.1 Data Splitting Strategies.... 132
5.6.2 Training Techniques and Algorithms Applied.... 133
5.6.3 Evaluation Metrics and Performance Analysis.... 134
5.7 Hyperparameter Tuning and Optimization Strategies.... 137
5.7.1 Grid Search Approach.... 137
5.7.2 Random Search Methodology.... 138
5.7.3 Bayesian Optimization Techniques.... 138
5.7.4 Early Stopping and Regularization Techniques.... 139
5.8 Deployment of Intrusion Detection in SCADA Systems.... 141
5.8.1 Real-Time Integration and System Compatibility.... 141
5.8.2 Handling Streaming Data in Live Environments.... 141

5.8.3 Automated Alerts and Incident Response Mechanisms 142
5.9 Adversarial Testing and Security Validation 143
5.9.1 Crafting Adversarial Examples 143
5.9.2 Evaluating Robustness Against Evasive Techniques 143
5.9.3 Defense Mechanisms and Mitigation Strategies 144
5.10 Comparative Evaluation with Legacy IDS Methods 145
5.10.1 Metrics Used in Benchmarking 146
5.10.2 Comparative Analysis of Strengths and Weaknesses 148
5.10.3 Quantitative and Qualitative Assessment Outcomes 150
5.11 Summary and Future Research Directions 151
Chapter 6: Experimental Results and Analysis 153
6.1 Introduction 153
6.1.1 Overview of Data Presentation 154
6.1.2 Tools and Software Utilized 155
6.2 Experimental Setup 156
6.2.1 SCADA System Configuration 156
6.2.2 Machine Learning Environment in MATLAB 157
6.3 Dataset Overview 157
6.3.1 Source and Composition 158
6.3.2 Preliminary Statistical Analysis 158

6.4 Results from Traditional Intrusion Detection Techniques 159
6.4.1 Detection Rate and Performance Metrics 159
6.5 Machine Learning Model Results 161
6.5.1 Decision Trees: Performance Metrics and Visualization 161
6.5.2 Neural Networks: Convergence Analysis and Performance 163
6.5.3 Support Vector Machines: Decision Boundaries and Metrics 164
6.5.4 Ensemble Methods: Combined Model Performance 165
6.6 Comparative Analysis 166
6.6.1 Metric-Based Comparison: Traditional vs. Machine Learning 167
6.6.2 Visualization of Comparative Results 168
6.7 Anomalies and Outliers 172
6.7.1 Identification Techniques and Tools 172
6.7.2 Impact on Model Performance 173
6.7.3 Screenshots of Outlier Analysis 174
6.8 Adversarial Settings Results 177
6.8.1 Efficacy of Models against Adversarial Samples 177
6.8.2 Visual Representations of Attack Patterns 178
6.8.3 Screenshots from Adversarial Test Cases 179
6.9 Real-Time Detection and Deployment Results 180
6.9.1 Performance Metrics in Real-Time Environments 180

6.9.2 Visualization of Real-Time Data Streams 182
6.10 Time serious and feature selection 183
6.10.1 Time serious 183
6.10.2 Feature selection 184
6.10.3 Various metrics comparison 185
6.11 Conclusion of Results 187
6.11.1 Key Takeaways from the Experiments 188
6.11.2 Limitations and Challenges Encountered 188
6.12 AI-Driven Detection and Impact Assessment of Stealthy Deception Attacks (SDAs) in SCADA Systems for Water Treatment Plants 189
6.12.1 Dataset Overview and Preprocessing 189
6.12.2 AI Model Training and Validation 190
6.12.3 Detection of Stealthy Deception Attacks 191
6.12.4 Impact of Sensor Data Manipulation on System Stability 192
6.12.5 Evaluation of Real-Time Anomaly Detection Systems 192
6.12.6 Comparing with literatures 193
6.12.7 Result of Stealthy Deception Attacks (SDAs) 197
Chapter 7: Proposed SCADA Protection Framework 203
7.1 Introduction to the Protection Hub Concept 203
7.2 Overview of Stealthy Deception Attacks (SDAs) 205

7.3 Machine Learning Models in SCADA Protection Hub 205
7.3.1 Decision Tree Classifier 206
7.3.2 Classifier for Neural Networks 207
7.3.3 Support Vector Machine (SVM) Classifier 208
7.3.4 Random Forest Classifier 209
7.3.5 Voting Classifier Ensemble 209
7.4 Confusion Matrix Analysis Summary 210
7.5 Implementation Considerations 210
7.6 Conclusion 211
Chapter 8: Discussion 212
8.1 Revisiting the Research Objectives 212
8.1.1 Alignment with Initial Goals 212
8.1.2 Achievements and Milestones 212
8.2 Comparative Analysis with Existing Literature 213
8.2.1 Similarities in Findings 213
8.2.2 Divergences and Unique Insights 214
8.3 Time serious and feature selection 215
8.4 Research's Effects 217
8.4.1 Industry-Based Useful Conventions 217
8.4.2 Theoretical additions to the field 218

8.5 Guideline for Further Education 219

8.5.1 Possible Research Domains 219

8.5.2 Methodological Developments 220

8.6 Evaluating the Research Trip 220

8.6.1 Notable Realizations and Learnings 221

8.6.2 Person Development 221

Conclusion 223

References 224

Table of figures

Figure 1 SCADA Security 20

Figure 2 Vulnerabilities in SCADA Systems 23

Figure 3 Intrusion Detection System (IDS) 29

Figure 4 Signature-based vs. Anomaly-based Detection 32

Figure 5 Machine Learning-based Intrusion Detection Systems (ML-based IDS) 40

Figure 6 Framework Diagram 76

Figure 7 framework diagram 131

Figure 8 Performance Metrics from Traditional Intrusion Detection Techniques 160

Figure 9 Decision Trees 162

Figure 10 Neural networks 164

Figure 11 Support Vector Machines (SVM) 165

Figure 12 Ensemble methods 166

Figure 13 Bar Graph: Detection Rate Comparison 169

Figure 14 Pie Chart: False Positive Rate Comparison 170

Figure 15 Line Chart: Update Frequency Over Time 171

Figure 16 Heatmap: Confusion Matrix Comparison 172

Figure 17 Box Plot: Data Distribution and Outliers 175

Figure 18 Scatter Plot: Data Distribution with Potential Outliers 176

Figure 19 Histogram: Data Distribution with Outliers 177

Figure 20 A hypothetical heatmap might showcase the intensity or frequency of adversarial attacks across different regions of the input space. Brighter regions might signify areas frequently targeted by adversaries, revealing potential model vulnerabilities. 179

Figure 21 hypothetical heatmap showcasing the intensity of adversarial attacks across different regions of the input space 180
Figure 22 line chart visualizing the hypothetical real-time data stream of network packets processed over time..... 182
Figure 23 Time Series Analysis: Count of Intrusions Over Time..... 184
Figure 24 Feature Importance for Intrusion Detection 185
Figure 31 ML Models in SCADA Protection Hub 206

List of tables

Table 1. Major Cyber Incidents Targeting Industrial Control and SCADA Systems 73

Table 2: Comparative Metrics Overview 146

Table 3. Strengths and Weaknesses Comparison between Machine Learning-based and Traditional Intrusion Detection Systems 149

Table 4: Quantitative and Qualitative Comparison 150

Table 5 Performance Metrics of the Simulated Intrusion Detection System on SCADA Network Dataset 161

Table 6 Comparative Metrics for Traditional vs. Machine Learning Systems 168

Table 7 Commonly Used Statistical Measures for Outlier Detection 173

Table 8 Impact of Outliers on Various Algorithms 174

Table 9 Model Efficacy against Adversarial Samples 178

Table 10 Performance Metrics in Real-Time Environment 181

Table 11 Machine learning models perform on various metrics, assisting in the selection of the most effective and efficient intrusion detection system for SCADA networks. 187

Table 12. Performance Comparison of Machine Learning Models for Intrusion Detection in SCADA Networks 190

Table 13. Real-Time Monitoring and Anomaly Detection of Chlorine Levels in a SCADA Water Treatment System 191

Table 14. Real-Time pH Level Monitoring and Anomaly Detection in a SCADA Control Process. 192

Table 15. Flow Rate Anomaly Detection and System Response Performance in SCADA Network. 193

List of Abbreviations

SCADA	Supervisory Control and Data Acquisition
IDS	intrusion detection systems
IT	Information Technology
IP	Internet Protocol
ML	Machine Learning
MATLAB	Matrix Laboratory
ROC	Receiver Operating Characteristic
AUC	Area under the ROC curve
PC	Personal Computer
LAN	Local area network
IoT	Internet of things
PLC	Programmable Logic Controllers
OT	Operational Technology
AI	Artificial Intelligence
HMI	Human-Machine Interfaces
OCR	Optical Character Recognition
NIDS	Network-based intrusion detection system
HIDS	Host-based intrusion detection system
SVM	Support Vector Machines
PCA	Principal Component Analysis
MTU	Master Terminal Unit
RTU	Remote Terminal Units

DDOS	Distributed Denial of Service
DNP3	Distributed Network Protocol 3
IEC	communication protocols for intelligent electronic devices at electrical substations, supporting industry-standard protocols
OPC UA	OPC Unified Architecture, supporting industry-standard protocols
Modbus	client/server data communications protocol, supporting industry-standard protocols
FPR	False Positive Rate
FNR	False Negative Rate
SGD	Stochastic Gradient Descent
L1	Lasso
L2	Ridge
TP	True Positives
TN	True Negative
FP	False Positive
FN	False Negative
H1 and H2	hyperparameters
R1 and R2	Ranges
F	Performance
P(h)	probability distribution
GP	Gaussian process
μ	mean
K	covariance

$\alpha(h)$	acquisition function
Lv	validation loss
λ	regularization strength
Wi	the model parameters
Δt	a small time interval
D(t)	continuous stream
$D(t-\Delta t,t)$	The data from the time window
θ	threshold value
O(t)	the model's output
A(t)	the system triggers
I(t)	incident response
X	original input
M	model
C	classifier function
δ	perturbation
C(M(x))	correct classification
R	robustness
A	adversarial samples
T'	the augmented training set
DR	Detection Rate
IQR	interquartile range
TTPs	tactics, techniques, and procedures
IDPS	intrusion detection and prevention systems

BCM	business continuity management
IEC	International Electrotechnical Commission
IEEE	the Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization

Chapter 1: Introduction

1.1. Background and Context

Many industries such as water supply and energy production, have benefited greatly from the increase in Supervisory Control and Data Acquisition (SCADA) systems. In the past, different systems did not exchange data with each other through a network. Being separated from others kept the place quite safe. Thanks to digital advancements and greater use of real-time and remote data, SCADA systems need to work with other information-based systems and connect between each other (Sangoleye, 2024).

SCADA technology is responsible for governing key physical functions in vital sectors. They comprise, among other things, the creation of electricity, the water pressure in distribution systems and the management of public transit. Since SCADA systems are so important in modern operations, any problems with them may cause terrible accidents (Lu, 2024).

Moving from proprietary systems isolated from the Internet to IP-based interconnected systems has given us better efficiency, reduced expenses and improved monitoring. At the same time, because SCADA networks are so closely tied to other systems, they are vulnerable to a range of threats now affecting information systems everywhere. Angling from traditional IT systems, what's most important in SCADA is that the data remains accurate and accessible at all times. An unauthorized person who enters the system could modify data or stop important processes, leading to hazards, fatalities and major financial losses (Idima, 2025).

SCADA has many different risks. Examples are problems that happen inside a company, external factors, natural catastrophies and accidents with technology. Among these, cyber-attacks orchestrated by malicious actors have been receiving significant attention, especially post high-profile attacks such as Stuxnet, which targeted nuclear facilities in Iran (Jensen, 2024). The techniques employed in these cyber-attacks have ranged from using zero-day exploits to socially engineered phishing campaigns.

Typically, cybersecurity efforts have involved protecting the system “at the door” by focus Contemporary threats necessitate that security measures be more flexible. Ids is where things get more interesting. They scan for any raising alarms in the activities on the network or the system. If something suspicious is found, alerts are immediately sent, so it can be dealt with promptly.

Although traditional intrusion detection has proved effective, the particular requirements of SCADA—such as their unique protocols and instant operations—make applying these solutions difficult. In addition, as data grows rapidly and attack methods become more advanced, we require advanced technologies to better detect threats.

As a branch of artificial intelligence, machine learning can recognize patterns and find unusual events. Since it can study data, machine learning can detect the standard or normal operations in a SCADA system and identify outliers that alert to possible intrusions (Bhatt, 2024).

SCADA systems are crucial toward supporting critical infrastructure globally by controlling aspects such as the electrical grid, water systems and transportation networks (Sangoleye, 2024). Linking SCADA systems with online devices has allowed them to function more widely, but it has left them open to various advanced cyberattacks. These weaknesses make both national security

and public safety highly at risk which is why creating strong cybersecurity solutions is so important (Byres, 2013; Sangoleye et al., 2024).

Protection of SCADA networks nowadays mostly depends on intrusion detection systems (IDS), which can spot anomalous activity suggesting the existence of cyberattacks or system problems (Byres, 2013; Sangoleye et al., 2024). Not always able to identify fresh or complex threats are signature-based or rule-based intrusion detection systems. As a result of this, machine learning (ML) has garnered a lot of attention since it is able to analyze enormous amounts of data and discover minute signs of assaults without the need to be aware of their signature (Byres, 2013; Sangoleye et al., 2024).

According to Reddy (2024), the fact that SCADA network data is constantly changing and frequently calls for rapid detection by machine learning has resulted in difficulties for intrusion detection. These difficulties compel and require real-time analysis as well as the capability for models to adapt. Mesadieu (2024) and Madhu et al. (2025) found that the use of machine learning methodologies, in which attackers intentionally fool the system into missing their attempts, adds additional obstacles to the security of the system. Consequently, the combination of decision trees, neural networks, support vector machines, and random forests through voting, which is accomplished through the utilization of a multi-model approach, has the potential to enhance both the detection rate and the system's capacity to withstand evasion.

1.2.Problem Statement

SCADA systems are now essential for crucial sectors in infrastructure such as energy, water processing, transportation and manufacturing. Today’s industrial processes depend heavily on these systems that manage and track physical events. When these security systems fail, the results

can be significant economic damage, harm to infrastructure, harm to the environment and in some cases, people may die. The big issue is that critical systems are attacked now as much by cyber methods as by physical attacks (Cherdantseva et al., 2016).

The basic architecture of SCADA traditionally used separate networks, though that is now changing. With the demands of modern industry and the integration of Internet of Things (IoT) devices, SCADA systems have become more interconnected, often interfacing with enterprise IT networks and even the public internet (Zhang et al., 2018). Despite being essential for monitoring, operating remotely and optimizing business, this new network infrastructure has caused a huge increase in risks for cyberattacks.

There is no speculation here – attempted intrusions into SCADA networks are becoming more frequent and actual. The Stuxnet worm, found in 2010, marked a major change in cybersecurity by confirming that countries were attacking industrial control systems and SCADA for their political purposes (Madhu, 2025). After Stuxnet, many more malicious campaigns such as BlackEnergy, Industroyer and Triton showed just how interested threat actors became with this area (Lu et al., 2024).

While INT systems in enterprise IT networks succeed in preventing cyber threats, the use of similar technology in SCADA systems can be complicated. First, there are big differences in how SCADA systems and behaviors compare to those of standard IT networks. Routinely occurring traffic from SCADA devices may fool traditional IDS, causing it to mistakenly detect lots of harmless messages (Harisha, 2025). Additionally, many SCADA networks have older equipment that isn't very powerful, making it tougher to introduce full-featured IDS solutions.

Because SCADA systems are so important and unique, it is now necessary to develop specially designed, adaptable and efficient ways to catch intruders. Using machine learning which specializes in spotting differences, detecting unusual behaviors and adjusting, looks like a promising step. Yet, putting machine learning into use in the detection of SCADA attacks is complicated, so it is crucial to test and refine these techniques extensively.

The study introduces the SCADA Protection Hub System which combines ML features to make it easier to detect intrusions within SCADA areas. We review the outcomes generated by various classifiers and meticulously investigate the patterns seen in time series sets of intrusion events, this research aims to advance both the theoretical understanding and practical deployment of secure SCADA networks.

1.3.Objectives of the Study

Using machine learning methods, the main goal of this research is to improve how intrusions are detected in SCADA networks. The key role SCADA systems play in supervising vital infrastructural and industry-related processes means their security needs to be top priority. These specific objectives have been outlined to give this research effort a clearer and more direct course:

1. **Analyze the status quo of the industry.**
    - Reviewing published research on SCADA intrusion detection methods so as to recognize what challenges and gaps already exist.
    - To learn about the main machine learning tools in use and how they may support SCADA systems.

2. **Design and carry out machine learning models for use.**
    - For the creation of machine learning models that focus on intrusion detection specifically in SCADA systems.
    - Make MATLAB the key software for starting from developing to testing these machine learning models.
    - to judge the performance of various machine learning algorithms based on their accuracy, how quickly they detect and how often they make errors.
3. **Evaluate real-life traces of SCADA traffic.**
    - To gather and preprocess actual SCADA network traffic to support training and validation of our models.
    - The models will be tested using newly encountered threats to confirm their ability to detect them.
4. **Discuss and handle problems that arise or are present.**
    - Investigating potential problems that may come up when deploying machine learning in SCADA settings, specifically due to processing delays and the privacy of data.
    - To design methods that lower the threat of adversarial attacks affecting the suggested machine learning models.
5. **Improve System Adaptability and Capacity to Scale:**
    - Transferring and online learning practices will make sure the models have the agility needed to meet new and evolved threats.

- To allow for the growth of SCADA networks by scaling the solutions chosen without a major loss in performance.

6. **Facilitate Knowledge Transfer:**

   - To create user-friendly documentation and guidelines that aid industries in implementing the proposed machine learning solutions for their SCADA systems.
   - To organize workshops or webinars that engage stakeholders and provide hands-on experience with the developed models and tools.

In achieving these objectives, the study aims to offer a holistic approach to improving intrusion detection in SCADA systems. It is envisioned that the results will not only contribute to academic discourse but will also provide practical solutions that industries can adopt to safeguard their critical operations.

1.4 Significance of the Research

Because SCADA systems keep many vital infrastructures working, guaranteeing their security is very important today. They manage services that support everyday life, including electricity supplies, water delivery, public transport and production. An attack on these networks can bring economic instability and also threaten the lives of people. Focusing on improving SCADA system intrusion detection with machine learning is very important for several good reasons.

1. Because threats are becoming more sophisticated, traditional intrusion detection systems sometimes aren't effective enough. The variety in these threats, whether APTs or zero-day vulnerabilities, enables them to pass through most existing security measures. Because

machine learning can update itself with each bit of new data, it promises to find and deal with these advanced types of threats (Buczak & Guven, 2016).

2. When key infrastructure is successfully attacked by cyber-criminals, the results can be major economic losses. Any breach in a SCADA system can quickly stop factories from working, disrupt the power supply and lead to major expenses. Consequently, this work can reduce the possibility of economic losses due to improved intrusion detection.

3. For the sake of public trust and safety, SCADA systems manage important services people use regularly. If there are weaknesses, public safety might be put in danger due to contamination of water or failures in public transport. If we have strong intrusion detection, it will help keep essential services safer and more trusted.

4. While there is a lot of research on machine learning, less focus has been put on using it with SCADA systems yet. This research will contribute valuable knowledge to the nexus of machine learning and SCADA security, providing insights and methodologies that future researchers and practitioners can build upon.

5. Since IoT and Industry 4.0 are bringing more internet connections to important industries, SCADA networks are expected to grow in size and complexity. With the results from this research, security mechanisms can develop alongside changes in these systems.

6. Intrusion Detection in SCADA: Showing machine learning works well for security could help the industry accept this solution. As a result of better security, some organizations will invest in further protecting their systems.

1.5 Overview of Machine Learning in Intrusion Detection

In recent years, cybersecurity threats have gotten much smarter, harder to eradicate and more severe. Therefore, IDS must change from their old signature-based approaches to better, adaptive models. Since ML helps find meaning in large and complicated information, it is an effective approach to examine diseases in this area.

At the center of machine learning is its role in helping computers understand and make choices from historical data. Rather than being instructed line by line, ML systems use algorithms to review data, gain insight from it and then make an educated estimate on something in the world (Bishop, 2006). Because it can spot both patterns and unusual events in vast data sets, this ability excels in intrusion detection.

Supervised Learning in Intrusion Detection

A major application of machine learning in intrusion detection uses supervised learning. Phase one consists of training algorithms using a dataset where the answers to each point are available. In this case, the model is built on network traffic data, with each example assigned to the category of either normal or attack. One the model is properly trained; it will examine new data and determine if it is an attack.

For instance, Decision Trees, Neural Networks, and Support Vector Machines (SVM) have been extensively used for classification tasks in intrusion detection. SVM, in particular, has been popular due to its ability to handle large feature spaces and its effectiveness in classifying nonlinear data (Lu, 2024, Harisha et al, 2025).

Unsupervised Learning in Intrusion Detection

Unlike supervised learning, where responses are given by the trainer, unsupervised learningnal does not depend on such training data. This approach is helpful when detecting intrusions, since besides known attacks, real-world threats may not be well known. Clustering is often used when grouping data based on their resemblance. Identifying clusters of network behaviors allows us to notice any anomalies which may point to an intrusion. Both K-means and Hierarchical clustering are standard methods used in this area (Lachure, 2024).

Modern techniques for learning in intrusion detection equipped for adaptation

In this approach, a software agent acts repeatedly inside a predefined system to achieve the best results. Reinforcement learning can allow systems in IDS to adjust as times goes on. When the IDS is given feedback by the system on its actions, it uses that to improve its ability to spot unusual attacks.

Challenges and Factors to Think About

Though machine learning provides many benefits, there are obstacles to using it in intrusion detection. The main issue centers around how excellent and complete the training data is. If the samples in the training data do not accurately reflect real life, the model could give a lot of incorrect answers. Besides, adversarial machine learning involves adversaries using deliberately manipulated data to trick ML models.

In addition, ML models, mainly deep learning models, consume significant computing resources. It could be too difficult for resource-poor environments to implement such demand in real time.

Because machine learning can handle large amounts of information and find fine trends, it brings a unique view to intrusion detection. When cyber threats evolve, ML approaches and cybersecurity hand in hand become essential. However, it's crucial to approach this integration with a comprehensive understanding of both the capabilities and limitations of machine learning.

1.6 Limitations of the Study

Every kind of research contains some constraints that might limit how much you can learn from it. It is crucial to acknowledge and define these problems so that the findings are explained with complete transparency. In this study which seeks to advance intrusion detection for SCADA networks with machine learning in MATLAB, the following constraints have been recognized:

1. **Data Constraints:**
    - Sometimes, the study utilizes shared datasets or private ones from certain partners which may miss some types of SCADA traffic and attacks.
    - Data used by SCADA systems in real life may include sensitive information. For confidentiality and compliance with guidelines, this research might analyze data that has been sanitized or anonymized which could make it less thorough.
2. **Difficulties Arising from Tools and Platforms**
    - Since the study mainly uses MATLAB, it does not evaluate how other machine learning development platforms might improve the process.
    - Depending on how much computing power MATLAB offers, complex machine learning models can take increased training and testing time.

3. **What Limits Machine Learning Models**

    - Often, researchers concentrate on particular learning models and miss out on other useful algorithms.
    - Models in machine learning sometimes can be manipulated or attacked in unknown ways and our study does not address them all.

4. **Generalizability:**

    - Because the findings are based on the datasets used, they may not be useful exactly as they are for other SCADA systems and areas.
    - While the models are useful for current threats, it is uncertain whether they will protect against new threats.

5. **Time and Resource Constraints:**

    - Given the time-bound nature of most research projects, the study might not be able to extensively tune or evaluate every potential machine learning model.
    - Due to potential resource constraints, real-time or large-scale deployment and testing of the developed intrusion detection systems might be limited.

6. **Evolution of Technology and Threats:**

    - As technology and cyber threats evolve rapidly, the study's findings might be more relevant to the current state of SCADA systems and may require periodic updates or revisions for continued relevance.

1.7 Chapter Organization

Chapter 1: Introduction

This chapter sets the stage for the entire research, providing an overview of the primary research area. It covers the background and context, highlighting the importance of SCADA systems and their vulnerability to cyber threats. The objectives of the study are clearly defined, followed by the problem statement which elaborates on the research gaps and challenges. The significance of the research is discussed, emphasizing its relevance in today's digital age. The chapter concludes by listing out the limitations faced during the study, ensuring transparency in the research process.

Chapter 2: Literature Review

This chapter looks at and reviews all literature related to SCADA systems, intrusion detection and machine learning. It explains how SCADA security has evolved, describes many security threats and weak points and introduces modern approaches and methods to managing them. In addition, the chapter talks about various machine learning algorithms that could be used in the field of intrusion detection. By finishing the book, readers are expected to know the leading techniques and unresolved questions.

Chapter 3: Methodology

The researchers detail the approach and structure they will use. The first part of the chapter is about deciding on the selection process for SCADA datasets and how preprocessing was accomplished. It explains each type of machine learning employed, also pointing out the algorithms and their implementation in MATLAB and the reason for those decisions. This chapter also gives an overview of testing and validation steps to maintain reliable and valid results.

Chapter 4: Results

This chapter clearly presents all the important findings from the study. The findings from machine learning are shown in detail, including rates of correctness, false positives and important statistics. Extensive use of graphs, charts and tables can help show the data in a more understandable way. The chapter provides information in numbers on how well the suggested machine learning algorithms performed in SCADA intrusion detection.

Chapter 5: Discussion

After presenting the findings from the previous chapter, this section examines them qualitatively. Results are examined using the knowledge found in earlier research and existing theory. The authors analyze the outcomes in detail, check how they compare to already known remedies and check the functioning of several machine learning models. Problems encountered, surprising results and lessons learned are examined to give a thorough understanding of what the research means.

Chapter 6: Conclusion and Future Work

The final chapter wraps up the research, summarizing the main findings and their implications. Recommendations for industries and policymakers are provided, offering actionable insights for enhancing SCADA security. The chapter also reflects on the limitations and challenges faced during the study, suggesting potential areas for future research. By the end, readers should have a holistic understanding of the research journey, its contributions, and the path forward.

1.8 Conclusion of chapter

Because digitalization is developing at such a rapid pace, the security of SCADA systems that manage essential infrastructures has become of the utmost importance. It is necessary to have methods that are both better and flexible in order to identify illegal access because the complexity of cyber threats is increasing. Because of the introduction, we are aware of the reasons why this research is necessary, the difficulties that are currently being faced, and the overarching goals that we aim to accomplish. In order to guarantee that SCADA networks are dependable and secure, the authors have high hopes that machine learning will eradicate the shortcomings that currently exist. In the introduction, it is emphasized that the findings of this research have larger implications for a variety of institutions, including government officials, the general public, and many companies. The study admits its limitations and instructs that this research be interpreted within the context in which the findings were attained. This is although the study covers a broad subject despite its scope. To accomplish this, the research discusses its objectives in the chapters that are to come and seeks to contribute to the enhancement of SCADA security, both for researchers and for applications that are now in use. In the next chapters, we will investigate the many types of literature, the methods that are utilized, and the innovative approaches that are established to achieve the objectives of the research. Rather than merely investigating SCADA and machine learning, this work demonstrates the collaboration that exists between the academic community and the business world in terms of safeguarding our most vital technological systems.

Chapter 2: Literature review

2.1 Historical Evolution of SCADA Systems

Over the course of many years, Supervisory Control and Data Acquisition (SCADA) systems have been an integral component in the automation and management of a wide range of industrial processes. Their development is a demonstration of the progress that has been made in technology as well as the unrelenting pursuit of efficiency in the operations of industrial facilities.

2.1.1 Early Origins and Development

SCADA systems can be traced back to the early 20th century, when companies relied on relays and timers to manage machinery and operations. This development led to the development of SCADA systems. These were manual systems that required a significant amount of labor and did not have the ability to scale.

In the 1940s and 1950s, with the advent of the electrical relay, the control mechanisms of industries underwent a transformation. Telemetry, the practice of collecting data over long distances, began gaining traction, especially in the utilities sector, like water and electricity distribution. Early telemetry systems used telephone lines and radio waves to transmit data (Byres, 2013).

The 1960s marked a significant turning point. The introduction of the mainframe computer provided industries with the ability to gather, analyze, and respond to data in real-time. These mainframes, however, were prohibitively expensive and required specialized environments, limiting their widespread adoption. But as technology progressed, the mainframes gave way to mini-computers in the 1970s. These mini-computers, more affordable and versatile than their

predecessors, allowed for more industries to integrate computerized control into their operations. It's during this period that the term "SCADA" was coined (Lachure, 2024; Galloway & Hancke, 2013).

With the onset of the 1980s, SCADA systems witnessed a significant leap with the advent of the personal computer (PC). Because PCs are both affordable and easy to use, they contributed to making computers available to many people. New software for SCADA was created for these platforms so that users could take advantage of better interactivity and visualization. Systems began to link more closely thanks to the use of Local Area Networks (LANs) which appeared in the early 1980s (Lu, 2024; Stouffer, Falco, & Scarfone, 2011).

At the beginning of the 1990s, the Internet increased the ease of sharing information over the web. With Internet communication, SCADA systems started forming networks that cover widely separated areas. Now, these systems could be used to watch over processes and make changes anywhere globally. But, this merging of ideas led to some negative results. Although it provided great freedom and control, SCADA systems were left open to many cyber-based threats, a situation that has not improved much ten years later (Lachure, 2024; Weiss, 2010).

SCADA technology has made great progress over the last couple of decades. Functions in systems are now closely linked, operating in a range of different industries. Contemporary SCADA systems function with fast communication, ongoing processing and modern graphic interfaces. Moreover, using IoT devices has allowed SCADA systems to become more independent and adjustable than they used to be (Lachure, 2024; Clarke & Reynders, 2019).

2.1.2 Transition to Digital SCADA Systems

The journey of SCADA systems from analog and relay-based systems to sophisticated digital platforms has been a testament to the transformative power of technology. Switching from traditional SCADA systems to digital options greatly improved the efficiency, dependability and integration of industrial processes.

In the past, industrial control systems largely depended on analog techniques using pneumatic and relay measures. Even though these systems were effective when first used, they had problems with flexibility, scaling up and efficiency. It used to be that changing the system meant literally rewiring its control panels which was a slow and prone to errors (Langill, 2012).

The 1960s saw the start of the rise of technology. With mainframe computers available, industries realized they could control their processes with computers. Because data processing was done centrally, mainframes simplified the work involved when analyzing various sources. Nevertheless, being large, costly and requiring a lot of infrastructure, these systems were unsuited to SCADA applications at a large scale (Lewis, 2011).

SCADA systems experienced major changes due to the introduction of microprocessors in the 1970s. Because of their small size, power and affordability, these devices could be fitted into a range of industrial instruments, sparking the growth of Programmable Logic Controllers (PLCs). Unlike earliest relay models, PLCs are programmable and can be used for many types of control tasks. Because records were digital, researchers could now collect, process and store data more accurately (Radvanovsky & Brodsky, 2013).

It was in the 1980s and 1990s that PCs were added to SCADA systems. Making processes more streamlined was possible because PCs introduced user-friendly interfaces, graphic means to view data and faster computing than required before. Easier connection was possible with LANs and

later on, using Internet protocols allowed for more communication and managing things from afar. At this time, specialized computer technology was developed to enable round-the-clock data monitoring, pattern analysis and alarming the user (Weiss, 2010).

With the rise of digitalization, SCADA started to be connected more closely with other IT systems, making it difficult to tell apart traditional IT and OT. With the rise in SCADA systems' integration, their risk of being targeted by cyber threats rose, so attention to cybersecurity was renewed.

Over the past few years, the Internet of Things (IoT), artificial intelligence (AI) and cloud computing have brought the SCADA and industrial sectors close together. Older SCADA systems remained in isolation, but today's tools are connected online and help drive industrial changes (Herrmann, 2018).

2.2 SCADA Security Concerns

Over the years, the administration of water treatment facilities to the control of electrical grids has shown that the usage of SCADA systems is absolutely needed for a great range of vital operations. Conversely, these systems grow increasingly prone to security breaches as they get more linked with bigger digital networks. Given its prominent importance in critical infrastructure, the very qualities that make SCADA so valuable—its adaptability, centralization, and connectivity—also make it more vulnerable to hazards. Figure (1) demonstrates the principle of supervisory control and data acquisition (SCADA) security that constitutes the overall framework of solving the problem of security of the elements of SCADA systems against cyber threats and attacks. The system is usually divided into three major layers as the Supervisory Layer which has servers and central operating stations, the Control Layer which has programmable logic controllers (PLCs) and remote-control units (RTUs) and the Field Layer which has sensors, actuators and physical

devices. The SCADA security is designed to guarantee data confidentiality, integrity, and availability through such technologies like firewalls, intrusion detection systems, encryption, and access control policies. The figure also demonstrates the way the components interact both on industrial and IT networks, which emphasize the role of breaking networks into segments and protecting communication channels against attacks by external and internal elements.

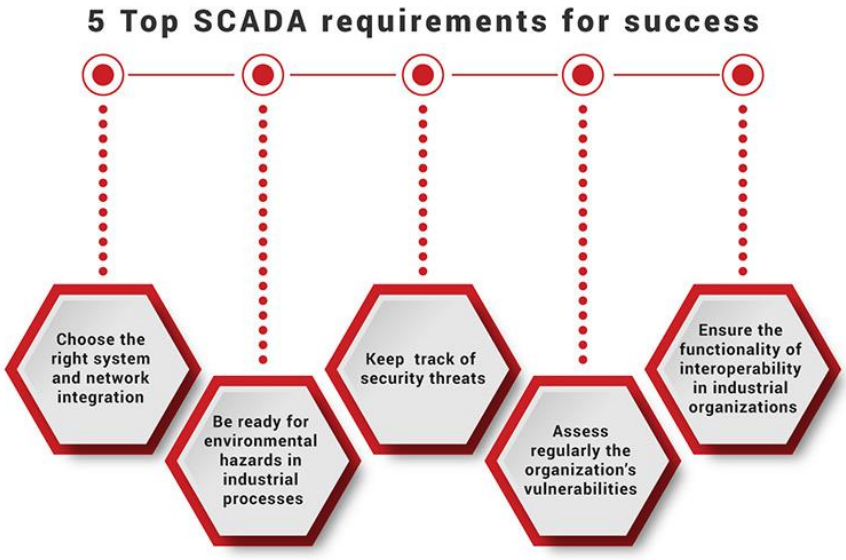


Figure 1 SCADA Security

2.2.1 Importance of Security in Critical Infrastructure

Understanding the significance of SCADA within critical infrastructure is crucial. To put it succinctly, critical infrastructure refers to the fundamental facilities, networks, and systems that ensure the smooth functioning of our societies and economies. This encompasses areas as diverse as transportation, power generation, health services, and more. Any compromise to these vital systems can lead to broad societal and economic upheavals.

In many of these sectors, SCADA systems lie at the core. They serve as the nerve centers, diligently monitoring and controlling intricate processes that maintain the pulse of daily life. For instance,

consider a bustling metropolis. The synchronized dance of its traffic lights, ensuring the ebb and flow of traffic, often depends on SCADA systems. Similarly, SCADA ensures that when residents turn on their taps, they receive clean water, and when they flip a switch, the lights come on. In such a connected ecosystem, these systems were engineered with a focus on operational efficiency and reliability, rather than security against sophisticated cyber threats (Anderson & Fuloria, 2010).

However, the implications of an unsecured SCADA system within such a framework are grave. Cyberattacks targeting SCADA could lead to widespread service disruptions. Imagine, for a moment, a scenario where a city's electrical grid is compromised. The resulting blackout could paralyze hospitals, render communication networks useless, halt public transportation, and plunge the entire city into chaos. Such disruptions can do more than just cripple an economy—they can endanger lives. And beyond mere disruptions, there's a more sinister possibility: malevolent actors might manipulate a compromised SCADA system to cause tangible, physical harm. In an industrial setting, equipment could malfunction, leading to accidents, environmental damage, or even loss of life (Mansfield, 2014).

Given these stakes, the focus on securing SCADA systems in critical infrastructure isn't merely a matter of operational continuity; it's a matter of public safety, economic stability, and, in many cases, national security. The junction of SCADA with contemporary digital technology offers both possibilities and problems. Leveraging data analytics, remote monitoring, and other tools, the chance is to maximize these critical roles even more. The challenge, then, is making sure we are simultaneously safeguarding these systems from the increasing amount of cyber-attacks while we are enjoying these advantages (Weiss, 2016).

2.2.2 Common Vulnerabilities in SCADA Systems

The change of SCADA systems from isolated, stand-alone operations to networked, internet-facing architecture has opened a plethora of benefits. These benefits especially help with accessibility from a distance and efficiency. Conversely, this development has unintentionally made these systems susceptible to a range of cyber vulnerabilities previously less of an issue.

As shown in Figure 2 below, the following are the most notable points that can be utilized to compromise the security of supervisory control and data acquisition (SCADA) systems. The system is usually composed of multiple layers that are linked together such as field sensors and actuators, programmable logic controllers (PLCs) and remote terminal units (RTUs), SCADA servers and human operator interfaces (HMIs) all the way to enterprise networks. The illustration indicates that all these layers have the possibility of vulnerability, including the absence of encryption in communication protocols, the physical vulnerability of field devices, the ease at which software vulnerabilities are exploited or the use of older operating systems, as well as the bad isolation of information technology (IT) and operational (OT) networks. The figure also reveals that the above vulnerabilities may be compromised by using different kinds of attacks, including denial-of-service (DoS) attacks, malware, man-in-the-middle (MIM) attacks, or even insider attacks by privileged users. The general idea behind the figure is the necessity to incorporate the Defense-in-Depth principle to make the system less vulnerable to system breaches and safeguard the critical infrastructure against cyber-attacks.

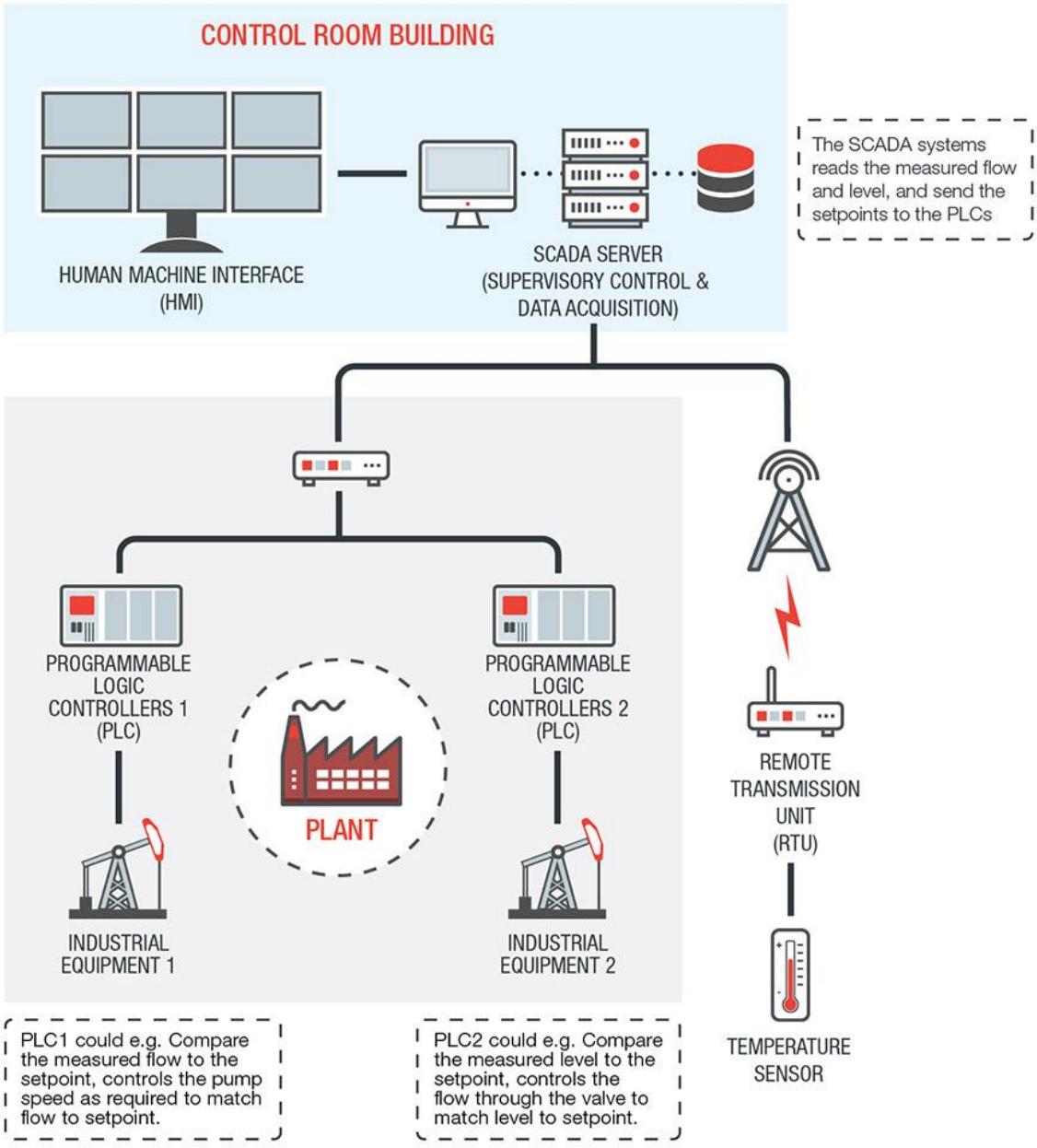


Figure 2 Vulnerabilities in SCADA Systems

Throughout their life, design goals for SCADA systems have mainly centered on having them be reliable, efficient and provide real-time responses. In the past, cybersecurity was rarely a priority when these systems were being built (Langner, 2011). Because SCADA systems have an old design, they now come with certain risks. Some cases arise when systems are vulnerable because they don't provide solid authentication systems. Moreover, information shared between SCADA

devices may not be encrypted, so it could be grabbed by unauthorized users (Rockwell Automation, 2016).

A further challenge is that the infrastructure relying on old and outdated systems. Although SCADA systems are well known for their long life, they still usually use outdated software and hardware. While "if it works, leave it alone," may be true in some areas, it can be very risky when it comes to cybersecurity. Stouffer, Pillitteri and Lightman (2015) argue that older systems often have security holes and may be unsuitable for the latest safety software. I am also worried about how easily these systems might be misused.

Connecting with the system if you're not close by is also challenging. Although remote access to SCADA systems is very convenient, it has its flaws. Without solid security for these remote areas, they might open a way for cyber attackers.

This situation is even more complex as SCADA systems depend on communication methods not widely used in other fields. Since there is not much testing of these custom processes, it's possible that unexposed vulnerabilities could still exist (Morrise, 2015). Many SCADA systems face both threats from outside as well as attacks from within their organization.

Finally, supervisory control and data acquisition (SCADA) systems aren't without their flaws, which stem mostly from their antiquated design and development, even though they're crucial to the management of modern infrastructure and the delivery of services. These can only be tackled with in-depth expertise and a multi-pronged strategy that prioritizes both technology and people.

2.2.3 Cyber Attacks on SCADA Networks

In the age of the internet and digital interconnectedness, the risk of cyberattacks on critical infrastructures has escalated. SCADA networks, which underpin much of our critical infrastructure,

have proven to be enticing targets for cybercriminals, hacktivists, and nation-state actors. Several attacks over the past years stand out due to their sophistication, the potential threat they posed, and the revelations they provided into vulnerabilities of SCADA systems.

The most widely recognized attacks that targeted a SCADA system is the Stuxnet worm, discovered in 2010. The meticulously crafted malware primarily aimed at Siemens' industrial control systems utilized in Iranian nuclear facilities. What set Stuxnet apart from other forms of malware was its objective – it was designed not just to infiltrate systems, but to cause tangible, physical damage. It covertly made the nuclear facility's centrifuges spin erratically, severely hampering their functionality, while simultaneously ensuring that the systems' displays read as though operations were normal. This attack was groundbreaking. For many security experts, it represented a turning point, highlighting the blurred lines between digital threats and their real-world repercussions (Zetter, 2014).

Yet, Stuxnet was not an isolated event. In 2015, another notable cyberattack was aimed at a critical utility, this time in Ukraine. Just days before Christmas, parts of Ukraine were plunged into darkness. A sophisticated cyberattack compromised the nation's power grid, resulting in blackouts and leaving approximately 230,000 residents without electricity during a notably cold period. This wasn't merely an act of digital sabotage; it had immediate, tangible consequences for the residents affected. Subsequent investigations revealed the use of malware known as BlackEnergy to gain entry into power distribution centers. This breach was not just a demonstration of vulnerabilities in SCADA systems but also raised geopolitical tensions, given the evidence suggesting that the Russian hacking group Sandworm was involved (Greenberg, 2017).

As a result of these and other attacks, it is of the utmost importance to immediately safeguard SCADA networks. In addition to this, they draw attention to the wider repercussions that

cyberattacks can have in today's interconnected society. Not only do these incidents constitute a threat to data or services, but they also have the potential to cause physical injury, disrupt lives, and even be exploited in bigger geopolitical confrontations.

2.2.4 Modern-day SCADA: Features and Advancements

Modern SCADA systems not only capture the benefits of digital transformation but also provide a range of features made feasible by current technology innovations. This is a big change from their forebears, who were like-minded. The contemporary SCADA system is a multifarious tool used for data analytics, optimization, and security, not only for process control.

One of the most important changes that has happened in present SCADA systems is integrated system architecture. This change entails the combining of several parts into a one whole body. SCADA components including data servers, communication servers, and Human-Machine Interfaces (HMIs) used to run historically either separately or independently of one another. SCADA systems are now offered as integrated packages, which simplify processes and ensure a continuous data flow (Mitchell, 2017).

Modern SCADA systems are designed to manage the enormous data creation produced by businesses. Furthermore, included in these systems are real-time data processing and analytics. The ability of these systems to process data in real time helps the operators to get fast insights. Modern SCADA systems also have embedded analytics tools, which let them foresee system faults, maximize operations, and maybe offer prescriptive solutions (Lee, Bagheri, & Kao, 2015).

Modern human-machine interfaces (HMIs) featured in SCADA systems offer chances for sophisticated visualization. Operators can access detailed and interactive dashboards that also

provide a whole picture of the system. Pan, Jain, and Paul (2019) claim that this not only facilitates more rapid identification and resolution of issues but also helps to better manage processes.

Cloud technologies and the Internet of Things (IoT) have made SCADA systems more linked than they have ever been in past years. Furthermore, growing in importance are remote access and connectivity. Large geographic areas can be covered with sensors and actuators, and data might be available almost from any point. Zhong, Newman, and Huang (2016) claim that this has especially helped in the era of Industry 4.0 since it has made remote monitoring and maintenance feasible.

SCADA systems grew more vulnerable to cyberattacks as their links grew as well. This underlines the need of cybersecurity. Given this, modern SCADE systems provide some kind of built-in cybersecurity protection. Among these defenses include encrypted communications, many factor authentication, intrusion detection systems, and regular security upgrades. Manufacturers and companies are also becoming aware of the need of following the best practices for cybersecurity, claims Stouffer, Pillitteri, and Lightman (2015).

Modern SCADA systems are designed to be compatible during the design phase with a broad spectrum of protocols and standards. This ensures that they can quickly interact with a range of hardware and software, therefore enabling companies to choose components most appropriate for their needs (Galloway & Hancke, 2013).

Scalability: Their control needs will change as these industries keep expanding and evolving. Two traits of current SCADA systems are scalability and modular design. This suggests, according to Clarke and Reynders (2019), companies are able to extend their control systems without suffering from downtime or major overhauls.

Ecological Development and Energy Conservation: Modern SCADA systems are also built with energy management in mind in view of the rising worldwide focus on sustainability. They can even interact with renewable energy sources (Wang, Chu, & Werth, 2018), evaluate patterns of energy consumption, enhance operations to lower energy usage, and thus optimize operations.

2.3 Intrusion Detection: An Overview

Since technology is being used more widely in many areas and cyber-attacks are on the rise, strong security is needed now. Securing networks and information systems often depends on the practice of intrusion detection. Intrusion detection systems (IDS) are used to watch networks and alert users when they find any suspicious or harmful activity. Due to the rise of sophisticated and more frequent attacks online, organizations rely on Intrusion Detection Systems.

2.3.1 Basics of Intrusion Detection Systems (IDS)

Intrusion Detection System (IDS) is a tool used to watch network activity and spot suspicious actions, violations or breach attempts and provide a report to a management console. IDS mainly helps administrators detect issues in time so that security can be prevented before serious harm happens.

Figure 3 shows the overall design and operation of an intrusion detection system (IDS) that is deployed to defend against cyber attacks on SCADA networks and systems. The figure illustrates that the IDS is a monitoring and analysis interface between the different components of the network, which receives input of various sources of information including network packets, system logs and control messages of PLCs or RTU. This data is then processed by the system applying different methods, such as signature-based detection, which applies similarity to known patterns of attack, or anomaly-based detection, which is used to monitor variations to normal system

functioning. The figure also indicates that an IDS may be network-based IDS that communicates network traffic or a host-based IDS that communicates activities in devices themselves. In case of suspicion, the system notifies the monitoring unit or control center to take necessary measures whether it is isolating the infected device or canceling communication. The general purpose of the figure is to show the essentiality of intrusion detection system to improve the security of SCADA systems by identifying the attacks at an early stage and investigating its origin to reduce the effects of attacks on the critical infrastructure.

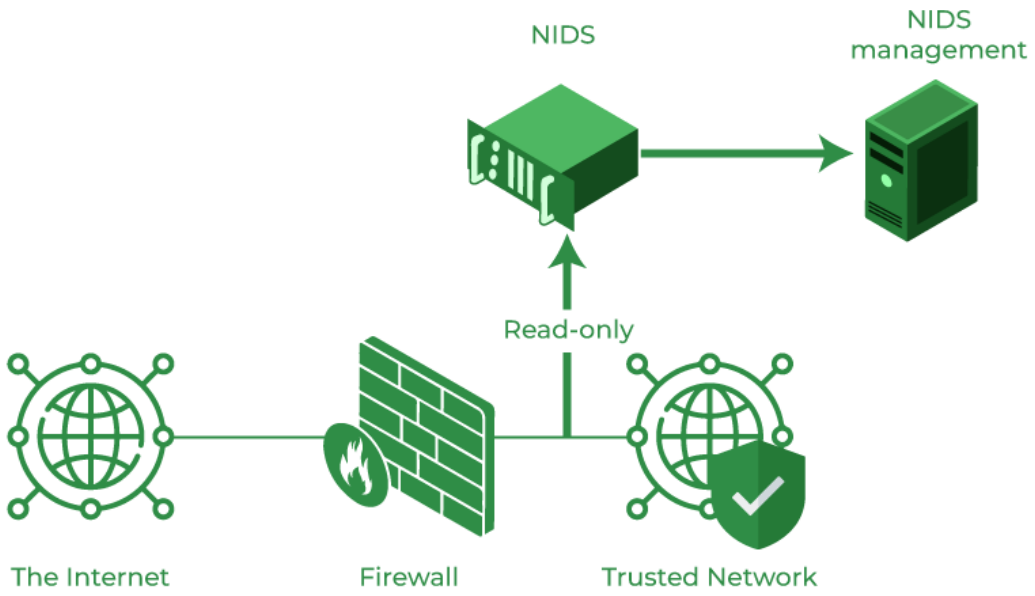


Figure 3 Intrusion Detection System (IDS)

There are mainly two types of intrusion detection methodologies:

Network-based IDS, or NIDS, examine incoming network traffic to spot any odd trends. Usually at the edge of the network, a NIDS is positioned strategically at a point within the network and looks at all traffic entering and leaving the network (Scarfone & Mell, 2007).

Host-based IDS, or HIDS, run on individual hosts or devices across a network. It solely tracks incoming and exiting packets from the device; should suspicious activity be found; it will notify the administrator or user. HIDS emphasizes system-specific hazards and could be able to identify if an assailant is trying to take advantage of a vulnerability found in a particular system or application (Bace & Mell, 2001).

Furthermore, separating the operational processes of IDS is depending on their detection strategies:

Using pre-defined patterns called signatures, this kind of signature-based detection seeks possible hazards. Every signature matches a certain set of characteristics connected to known harmful data or behavior. The IDS compares network traffic against these signatures and, should a match be discovered, an alarm is produced (Bace & Mell, 2001).

An anomaly-based detection system first creates a baseline of normal network activity rather than depending on known patterns of malicious conduct. It then constantly watches network behavior against this baseline, alerting any differences as possible hazards. Anomaly-based detection is adept at discovering new, previously unknown threats but can be prone to false positives (Axelsson, 2000).

2.3.2 Signature-based vs. Anomaly-based Detection

In the realm of cybersecurity, intrusion detection remains paramount, especially when cyber threats continually evolve in both frequency and sophistication. Two methodologies — signature-based detection and anomaly-based detection — have gained prominence in the intrusion detection domain. Each approach, while effective in its own right, offers distinct advantages and challenges.

Signature-based detection, often termed misuse detection, operates on a foundational premise. It matches patterns in network traffic to predefined signatures, which are sets of attributes tied to recognized threats. This method's strength lies in its precision. When there's a signature match, it can accurately pinpoint the threat, providing clarity for security professionals who then can act decisively. The efficiency of this system is noteworthy; it swiftly scans vast data swathes by merely matching against its signature database (Liao, Vemuri, & Wang, 2013). However, it's not without limitations. Signature-based detection is intrinsically confined to known threats. If an exploit is new or slightly varies from its database, it remains undetected. Furthermore, its effectiveness wanes without regular updates to its signature database.

Figure 4 is a comparison between two major categories of intrusion detection systems (IDS) applied in the network security: signature-based detection and anomaly-based detection. The figure usually shows a diagram or a table with a comparison of the functioning of each system, its features, and constraints. Data In signature-based detection, the system uses a database of known patterns or "signatures" of past attacks and compares them with current network data. In case a match is identified, it launches a threat notification. This form of IDS has a high level of accuracy and can detect known attacks very fast, however it cannot detect new and hitherto unreported attacks. Anomaly based detection, however, is based on developing a model of the normal behavior of the system or network and subsequently observing any deviation to that behavior as a possible attack. This kind of IDS is defined as sensitive to changes, and thus it can detect a new attack (zero-day attacks), but this IDS can have more false alarms because of its high sensitivity to changes. The entire figure illustrates that the combination of the two techniques is an efficient method of establishing a compromise between the accuracy and threat predictability to increase the efficiency of intrusion detection systems in areas like sensitive SCADA systems.

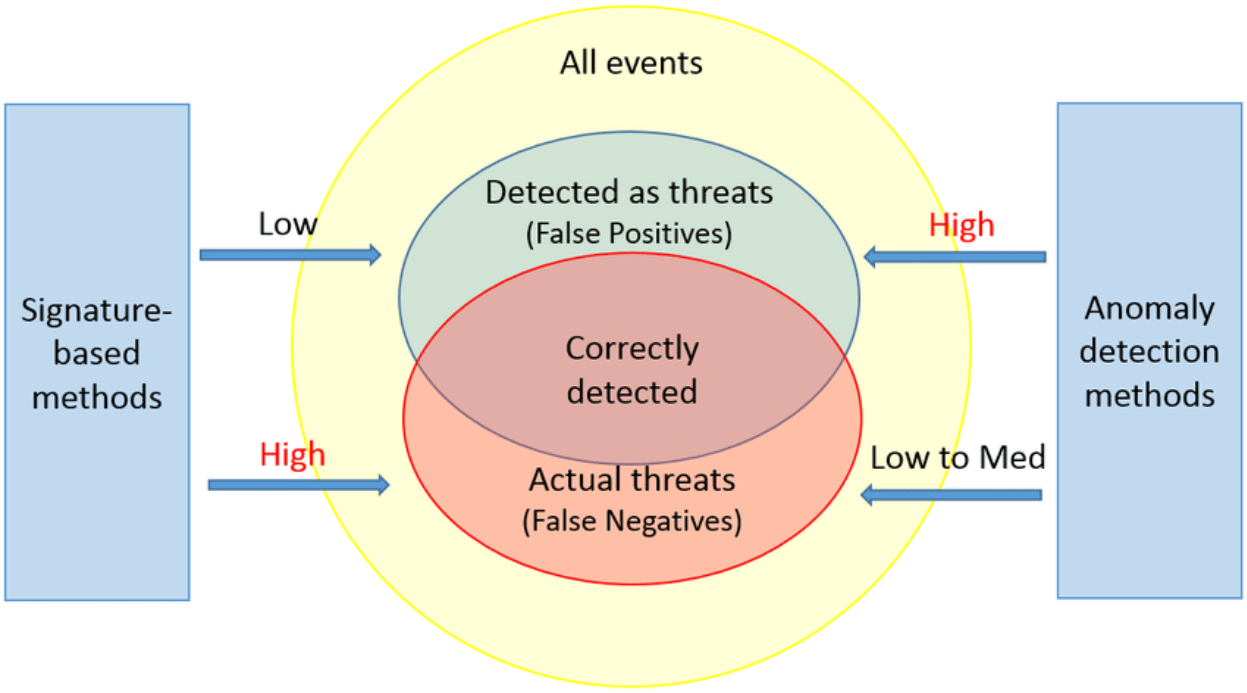


Figure 4 Signature-based vs. Anomaly-based Detection

On the other hand, anomaly-based detection contrasts sharply in its approach. It starts by observing network traffic over a set period to establish a 'normal' behavior baseline. Any subsequent deviation from this baseline, regardless of whether it's malicious or benign, is flagged. This approach's most significant advantage is its potential to unearth unknown threats, providing a safeguard against novel or zero-day exploits. It adapts readily to changing network environments, reflecting evolving user behavior patterns (Sommer & Paxson, 2010). Yet, challenges persist. Defining 'normal' is often intricate, and genuine behavior shifts can lead to false positives. The algorithmic complexity, combined with the computational resources required for continuous baseline updates, can strain organizations. Additionally, there's an inherent initial vulnerability, as a reliable behavioral baseline takes time to establish.

In essence, while both detection methodologies offer robust tools to identify cyber threats, they aren't without pitfalls. Often, the optimal strategy for organizations is a hybrid approach,

employing both signature-based and anomaly-based detection. This amalgamation maximizes their strengths while mitigating their individual weaknesses, ensuring a broader and more resilient defense against cyber adversaries.

2.3.3 Challenges in SCADA Intrusion Detection

SCADA (Supervisory Control and Data Acquisition) systems form the backbone of numerous industrial processes. These range from power generation, water treatment, to manufacturing. As such, their security isn't just about data protection; it's about ensuring the smooth operation of critical infrastructures that societies depend upon. The unique nature of SCADA systems, when juxtaposed with traditional IT environments, brings forth a set of distinct challenges when implementing intrusion detection. Let's delve deeper into these challenges.

1) Operational Continuity

SCADA systems prioritize availability. Any downtime can lead to significant economic losses, safety risks, and service interruptions. Traditional intrusion detection systems (IDS) designed for IT environments might take aggressive actions upon threat detection, like shutting down portions of the network. Implementing such measures in SCADA can lead to catastrophic results. Balancing between threat mitigation and ensuring continuous operations becomes a primary challenge (Byres, Franz, & Miller, 2004).

2) Legacy Systems and Proprietary Protocols

Many SCADA systems in operation today are based on legacy technology. These were designed decades ago when cybersecurity wasn't a primary concern. These systems often use proprietary or non-standard communication protocols. Traditional IDS, accustomed to standard internet protocols, can find it challenging to monitor and interpret SCADA-specific traffic effectively.

Adapting to this environment, understanding its nuances, and ensuring that alerts for malicious activities are neither missed nor misunderstood is a significant hurdle (Zhang, Song, & Chen, 2015).

3) Lack of Network Redundancy

Unlike typical IT networks that have redundancies built-in to handle failures, SCADA networks might not have such backup systems in place. If an IDS mistakenly identifies legitimate SCADA traffic as malicious (a false positive) and blocks it, it could disrupt the entire process, emphasizing the need for high accuracy in intrusion detection.

4) Limited Computational Resources

SCADA devices, especially those embedded in industrial equipment, often have constrained computational capabilities. They are optimized for their primary functions and not for running resource-intensive intrusion detection algorithms. As a result, deploying sophisticated IDS solutions on these devices without affecting their performance becomes a challenge (García-Teodoro, Díaz-Verdejo, Maciá-Fernández, & Vázquez, 2009).

5) Real-time Processing Needs

SCADA systems often require real-time monitoring and rapid response times. This demands that the IDS not only detect threats but also communicate them instantaneously to ensure timely interventions. Processing large volumes of network traffic in real-time, while maintaining accuracy, is inherently challenging.

2.4 Machine Learning in Cybersecurity

The cybersecurity landscape is in a perpetual state of evolution, with threats becoming more sophisticated and diverse. To keep pace with these threats and even preemptively detect or mitigate them, many organizations and researchers are turning to machine learning. Machine learning, a subset of artificial intelligence, has emerged as a transformative tool in the cybersecurity realm, providing automated, efficient, and dynamic solutions to a plethora of challenges.

2.4.1 Fundamentals of Machine Learning

In essence, ML is when computers using algorithms can improve performance with time, using given data, without direct instructions for each task. We will now study the important principles and ideas that make up machine learning and why it is related to cybersecurity.

Learning Paradigms

There are primarily three types of learning paradigms in ML:

1) Supervised Learning means the algorithm works with pairs of input and output from a labeled dataset. After training the model, it becomes able to predict what will happen in cases it hasn't seen before. This method is found in uses such as identifying spam emails, where every email is tagged with a label, for example, "spam" or "not spam" (Bishop, 2006).

2) With unsupervised learning, there are not labels for the data, as there are in supervised learning. It's easy to learn about the basic patterns or underlying groups in the data. Data points are often grouped in clusters, using cluster analysis which is an example of unsupervised learning.

3) Reinforcement Learning: Using this technique, the agent acts in the environment and learns from the feedback it is given, including penalties and rewards. Such a policy is often helpful when there is no certainty about outcomes and decisions must be taken in sequence, an example being network traffic routing (Sutton & Barto, 2018).

Features and Labels

For ML, the features are the inputs that the model studies and the labels are the results we hope to predict. As an example, determining if a file contains malware can use file attributes as features and each label would note if the file is malicious or not.

Model Training and Testing

As soon as the best machine learning algorithm is selected, it should be trained using a dataset. The model is evaluated on a different set of data after its training to see how accurately it works.

Evaluation Metrics

Accuracy, precision, recall and the F1 score are some of the ways used to measure model performance. Because the cost of errors in cybersecurity can be significant (for example, wrongly identifying a safe action as unsafe), measuring false positive and false negative rates is very important.

Relevance to Cybersecurity

The ability of ML algorithms to pick out patterns makes them valuable contributors to cybersecurity. With access to a wide range of data, these algorithms can detect unusual activity, “find” signs of familiar malware or even predict upcoming software problems (Jordan & Mitchell,

2015). In addition, because ML can adjust to emerging threats, the cybersecurity system always stays strong and up-to-date.

2.4.2 Application of Machine Learning in Cyber Threat Detection

Machine learning (ML) now plays an important role in cybersecurity, mainly when it comes to detecting threats. Because cyber-attacks are becoming more sophisticated, the traditional ways we use to detect them often miss them. Thanks to machine learning which looks at huge data sets, finds regularities and adjusts on its own, it is a promising approach. Here, we look at how machine learning helps find cyber threats.

1) Anomaly Detection

One of the most prominent uses of machine learning, cyber threat detection puts anomaly detection in the spotlight. The purpose of evaluating large datasets using anomaly detection methods, which are frequently unsupervised, is to establish a baseline for "normal" behavior. As per the research conducted by Chandola, Banerjee, and Kumar (2009), any variation from this set is taken into consideration as potentially signaling cybercrime or hostile conduct. One example would be a dramatic increase in network traffic, which could indicate a Distributed Denial of Service (DDoS) attack. Using machine learning techniques, such anomalies can be quickly identified.

2) Phishing Detection

phishing remains a major risk because dishonest people make these attempts to trick consumers into sharing personal information. It’s possible to develop algorithms that tell apart real websites from phishing websites by spotting small differences. Sahingoz, Buber, Demir and Diri (2019) claim that these approaches consider aspects such as URL patterns, website structure and what is on the site. With this setup, detecting phishing attempts is simpler and easier on a higher scale.

3) Malware Classification and Analysis

Since malware keeps developing, standard antivirus software cannot keep up. Machine learning is a helpful tool that marks dangerous software so it can easily be identified. By checking the code and activities of a file, tools use AI to figure out its threat level. According to Kolosnjaji, Zarras, Webster and Eckert (2018), machine learning variants called deep learning can detect fresh malware or slight updates with still high accuracy.

4) Network Intrusion Detection

An important component of detecting attacks is letting network intrusion detection systems study network traffic with machine learning. Vincent, Fulp and Greene (2011) explain that machine learning models can detect and block attempts at network intrusions using packet data, different protocols and network activity. Rules don't always catch these types of attacks.

5) Predictive Analysis

Identifying known risks is not the only use for machine learning; it is also used for predictive analysis. The accuracy of past security breach and cyberattack history analysis by machine learning algorithms determines the accuracy of their future predictions. In the view of Garcia, Zavarsky, Lindskog and Ruhl (2014), employers are able to take steps in advance to strengthen their defenses, thanks to the ability to predict breaches.

2.4.3 Strengths and Weaknesses of ML-based IDS

Modern cybersecurity has undergone a sea change with the introduction of ML-based Intrusion Detection Systems (IDS) to the ever-expanding cyber ecosystem. While ML-based intrusion

detection systems (IDSs) are fantastic for spotting threats, it's important for security professionals and businesses thinking about using them to know the pros and downsides of these systems.

Figure 5 demonstrates how machine learning systems can be utilized in constructing intelligent intrusion detection systems to defend the networks especially industrial SCADA networks. The illustration indicates the fundamental steps of an ML-based IDS, which begins with the data collection of all kinds of sources, including network packets or system logs. The preprocessing stage involves data cleaning and transformation into a digital format that can be used in learning. The extraction of features is subsequently done to determine the most significant variables in the process of identifying normal activity and malicious activity. The figure indicates that the second stage is the training stage, in which machine learning models including Decision Trees, Support Vector Machines, Random Forests, or Neural Networks are trained on historical data on the normal behavior patterns as well as the attack patterns. The next phase is the detection/testing phase where the trained model is used to predict new data to establish whether there is a threat. Another way is to apply a continuous learning layer in order to update the model with new threats appearing. The figure indicates that application of machine learning techniques with intrusion detection systems yields more recognition of newer and advanced attacks and fewer false alarms as compared to the traditional systems. The entire figure can be interpreted as the current tendency to use artificial intelligence and machine learning to make the SCADA systems more resilient and safer to the inception of sophisticated cyberattacks.

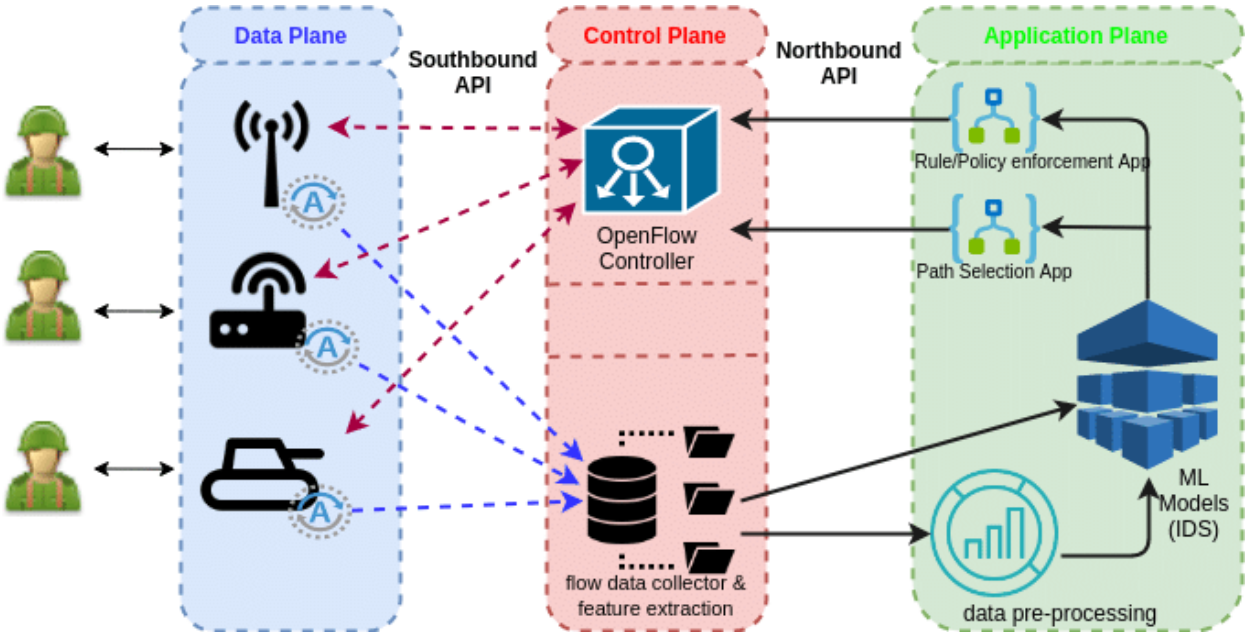


Figure 5 Machine Learning-based Intrusion Detection Systems (ML-based IDS)

One important benefit of machine learning-based intrusion detection is how flexible they are. However, machine learning systems adopt and grow over time, where traditional signature-based IDSs are fixed to specific criteria and often run out of date after a short period. In Laskov and Šrndić's (2011) view, ongoing learning allows them to handle new threats and maintain importance over a period. Both scalability and adaptability are strong characteristics of these systems. With more networks joining forces, the flood of data could easily grow to a point that is too much for security systems to handle. Machine learning is helpful in this field mainly because it can handle big chunks of data efficiently. So, they are able to discover threats more completely, particularly in complex networking situations.

Also, machine learning allows these systems to find zero-day attacks, that are not yet recognized by any other means. Depending mainly on standard indicators, conventional systems can easily be surprised by new styles of attacks. With this technology, it is also possible for such systems to pick out threats that have never been encountered in the past. This approach guards enterprises

against more possible threats of intrusions (Alazab, Venkatraman, & Watters, 2012). Moreover, thanks to certain machine learning models, people can take preemptive actions as well as dealing with attacks after they take place. These researchers indicate that the models make predictions about future threats by studying past occurrences and facts. Because of this, businesses can prepare their security systems in advance.

IDSs that are based on machine learning are extremely dependent on the data they use for training. It is possible for the subsequent analyses to be skewed, which can result in findings that are not reliable (Sommer & Paxson, 2010). This can occur if the training data is of poor quality, is not representative, or is even falsified. Not only that, but there is also the issue of excessive fitting. In the pursuit of precision, machine learning models may occasionally become overly adapted to the data they were trained on, which results in a reduction in their capacity to generalize and recognize threats in a variety of real-world settings. This is because the computational needs of machine learning, particularly the resource-intensive deep learning models, might be prohibitive for certain businesses, which in turn restricts the general use of these models.

As far as security is concerned, hostile attacks are a potential obstacle that must be overcome. At this time, sophisticated cyber attackers are developing inputs that are intended to trick and mislead machine learning models, hence causing these algorithms to incorrectly identify hostile activity. According to Biggio and Roli (2018), this vulnerability has the potential to dramatically jeopardize the integrity of the system it uses. Lastly, the fact that many machine learning models are black boxes can be a cause for concern. In situations when it is essential to have a solid grasp of the decision-making process, the absence of transparency and interpretability in machine learning decisions might present several issues.

2.5 Existing Machine Learning Techniques for SCADA Intrusion Detection

SCADA systems are critical in support of many vital infrastructures, so these systems are now heavily targeted by cybercriminals. So far this year, development has focused on integrating Machine Learning methods into securing SCADA networks from intrusions. Its goal is to look at the supervised learning techniques used to increase the safety of SCADA systems.

2.5.1 Supervised Learning Approaches

One example of supervised learning is training a model using labeled data, since this is a major part of machine learning. For this kind of learning, you receive both what needs to be learned and what the desired outcome is. When the model is trained, it is capable of projecting what the output will be for data it hasn't met before. This technique matters in SCADA intrusion detection since it enables using old attack details to guide models that avoid future problems.

2.5.1.1 Decision Trees

Decision trees are one of the main supervised learning approaches found in SCADA intrusion detection. Due to how they are designed, they are easy to follow, as they arrange data into groups per specific requirements and conclude the analysis. You can construct decision trees with SCADA systems to spot real network traffic from network attacks.

Maglaras and Jiang proved in their study from 2014 that decision trees assist in finding SCADA-targeted hack attacks. They were able to tell apart the two by having a model that could distinguish between regular use and attack. The capability of this approach to rapidly adjust to shifting network conditions is one of its advantages. This approach ensures strong intrusion detection even in dynamic SCADA environments, which is a significant advantage.

2.5.1.2 Neural Networks

Neural networks which look like the human brain, are built from interconnected nodes, known as neurons. Because they can detect difficult patterns well, they are a serious option to act as intrusion detection agents in SCADA networks.

In the year 2013, Hadžiosmanović, and et al performed research about using neural networks in intrusion detection for SCADA systems.

It was demonstrated in their study that neural networks can detect tiny errors in network data, helping them to ordinary problems with a very low rate of general errors.

2.5.1.3 Support Vector Machines

The work of SVMs is to search for a hyperplane that can cleanly divide a dataset into separate classes. SCADA intrusion detection with the SVM can determine whether network traffic is harmless or malicious.

Yang, Usynin and Hines (2006) found that with support vector machines (SVM), it is possible for SCADA systems to spot unusual behaviors. Research shows that SVM can successfully detect and label both normal and harmful activities in a SCADA system.

2.5.2 Unsupervised Learning Approaches

In contrast to supervised learning, unsupervised learning analyzes data without any labels to identify patterns. Because these methods find unusual or hidden signs even without knowing what a threat looks like, they are very helpful for monitoring SCADA. They are good because they are useful. There is a common method of using tools that do not need supervision to identify suspicious malware events not previously known.

2.5.2.1 Clustering Techniques

One type of unsupervised learning is known as clustering, and it involves identifying similarities between different data items and then utilizing those similarities to create separate categories. Using clustering, SCADA intrusion detection can identify and separate many different kinds of network traffic. Any unusual activity, compared to the usual picture, may signal that risks exist here.

Many organizations in SCADA employ the K-means technique, a clustering approach, very often. The dataset is used and for each point, its cluster is determined by placing it where the average distance from it to all other points is at a minimum, as long as the resulting number of clusters does not exceed a number K determined ahead. They applied k-means clustering effectively in 2016 to find the outliers within SCADA datasets. After reviewing the data, investigators found that data points that look very different from the average can help identify possible invasions. We were able to do this by creating multiple clusters of conventional network operations.

2.5.2.2 Principal Component Analysis

PCA which stands for Principal Component Analysis, is used to reduce the number of dimensions in a dataset. Here, the data is recalculated so that the input variables are replaced by a set of new variables called "principal components." Most of the data's changes are caused by these factors. To reduce the high number of variables in SCADA network traffic data, researchers use principal component analysis (PCA) for detecting intrusions.

In 2007, Huang and Leung used principal component analysis to check its ability to spot SCADA system attacks. Using PCA, we simplified the representation of the SCADA system's usual

activity. An attempt to steal information would be suspected if anything that differed greatly from this compressed representation was deemed strange.

2.5.3 Hybrid Models and Ensemble Techniques

Depending just on one machine learning model could not always yield perfect results for intrusion detection in the quickly growing sector of cybersecurity, especially when it comes to defending SCADA systems. This is the reason hybrid models and ensemble approaches have been under research. These techniques combine the benefits of several approaches to raise detection performance. These techniques help one to maximize the advantages of different methods while also reducing the shortcomings of any algorithm.

2.5.3.1 Hybrid Models

Hybrid Enhanced intrusion detection in SCADA systems is usually performed by means of hybrid models, which mix supervised and unsupervised learning approaches. Both supervised (which rely on labeled data) and unsupervised (which detects unseen patterns) techniques offer benefits; the idea is simple: by combining the strengths of both groups of techniques, the detection system can become more resilient and flexible. An unsupervised technique, clustering could be used in a hybrid model—for example, to split network traffic into different patterns. After then, depending on past labeled data, a supervised classifier—like a neural network—can be used to detect these patterns as either benign or malicious. To create a hybrid model for SCADA intrusion detection in 2015 Li, Yang, and Liu combined the K-means clustering method with a neural network classifier. The three researchers presented this model. They started by K-means segmenting network traffic patterns, then classified these patterns using the neural network. They were therefore able to greatly lower the amount of false positives at the same high detection rate.

2.5.3.2 Ensemble Techniques

The goal of ensemble approaches is to achieve a higher level of prediction performance than could be achieved by any one of the constituent algorithms when taken individually. This is accomplished by combining many learning algorithms. Within the context of SCADA intrusion detection, ensemble approaches have the potential to be very effective in reducing the number of false alarms and increasing the accuracy of detection levels.

It is commonly referred to as the "bagging" technique is a strategy that is applied rather regularly. Constructing a large number of subgroups from the initial data and then training a classifier on each of those subsets is the method that is utilized in this methodology to further the goal of intrusion detection, the researchers— Garcia-Teodoro, Diaz-Verdejo, Maciá-Fernández and Vázquez— investigated employing two ensemble techniques, bagging and boosting. Consultants looked at several approaches in addition to these. Based on their study, the researchers discovered that using ensemble approaches guarantees the SCADA system can react effectively to changing hazards and enhances detection. The reason is because these methods assist to generate better forecasts by leveraging the pooled knowledge of numerous models.

2.6 Evaluation Metrics for IDS

Making ensuring an IDS serves its intended use in cyberspace depends on proper performance assessment. By means of appropriate evaluation instruments, one can identify the essential elements of an IDS. For both SCADA and other critical systems, accurate evaluation on an IDS is absolutely crucial.

2.6.1 Accuracy, Precision, and Recall

An IDS is meant to spot hostile behavior right away and help to lower false alarm output. Evaluating intrusion detection system (IDS) models based on machine learning usually depends on accuracy, precision, and recall.

Accuracy is the count of accurate classifications displaying whether something is benign or dangerous. It can see for every data set entry the VALUE column. Using the mathematical ideas applied in the design:

$$Accuracy = \frac{True\ Positives + True\ Negatives}{Total\ Instances}$$

However, despite the reality that it provides an overall picture of the performance of the model, it is conceivable that accuracy is not totally accurate. The fact that one class (malicious/benign) considerably outnumbers the other is an indication that the dataset has a class imbalance. This is especially true in circumstances where the dataset has no class imbalance.

A concept that focuses on the accuracy of the positive predictions that are created by the model is called precision. Precision is also sometimes referred to as the positive predictive value. Precision is a concept that focuses on the accuracy of the positive predictions. Specifically, it calculates the percentage of cases that were correctly predicted to be positive (also known as True Positives) in relation to the total number of instances that were anticipated to be positive (which includes both True Positives and False Positives):

$$Precision = \frac{True\ Positives}{True\ Positives\ +\ False\ Positives}$$

It is vital for intrusion detection systems (IDS) to have a high precision since it implies that the model has a low false positive rate. This is because it helps reduce superfluous warnings that could overload system administrators.

The ability to recall (or detect): The ability of the model to recognize and record all of the positive examples is what you mean when you talk about recall. The ratio of True Positives to the total of True Positives and False Negatives (the occurrences of positive results that the model failed to identify) is what it is called:

$$Recall = \frac{True\ Positives}{True\ Positives\ +\ False\ Negatives}$$

A high recall is necessary for security applications such as intrusion detection systems (IDS) since it implies that the system is able to recognize a considerable number of genuine threats. This ensures that only a limited number of hostile activities are overlooked, which is why a high recall is essential.

In spite of the fact that Accuracy, Precision, and Recall separately provide valuable insights into the functioning of an intrusion detection system (IDS), it is frequently essential to view all three of them together in order to carry out a comprehensive review. A good illustration of this would be the possibility that an intrusion detection system (IDS) may achieve great precision at the expense of reduced recall, or vice versa. It is feasible for security analysts to prioritize one measure above another, based on the specific application and the severity of false positives in relation to false negatives.

For SCADA systems, where the repercussions of breaches that go undiscovered could be disastrous, it is of the utmost significance to strike a balance between precision and recall. Such behavior is extreme due to the seriousness of the results. This situation has led researchers to focus on ways of improving several parameters at the same time. Singh, Gupta and Singh (2018) studied different machine learning approaches for intrusion detection systems (IDS) in SCADA systems.

The researchers discovered that the models were not all the same. They chose to develop their skills in seeking the right balance between precise speech and remembering what to say.

2.6.2 False Positive and False Negative Rates

Faulty identification by intrusion detection systems (IDS) used for SCADA can bring very significant consequences. Both false positives and false negatives should be cut down by using an effective IDS. We also refer to false positives as false alarms. You can find out how trustworthy the model is and how its results might change the system's performance by assessing the rates.

Known as the False Positive Rate, FPR or briefly the False Alarm Rate: The False Positive Rate is a number that shows out of all the harmless activities, how many are mistakenly labeled dangerous. Another way to say this is that it is the "False Alarm Rate." Here's how you might say it in math terms:

$$FPR = \frac{False\ Positives}{False\ Positives\ +\ True\ Negatives}$$

The presence of high FPRs in an intrusion detection system (IDS) is troublesome, especially in applications that involve SCADA. It is potential for a system to produce "alert fatigue," which occurs when operators get desensitized to the numerous sirens and may fail to spot an actual threat. This can happen when the system is used more frequently. This is something that may take place if the system consistently produces false warnings from time to time. Additionally, responding to false alarms can divert resources away from more important concerns and may even result in wasteful downtime for the system if operators shut down areas of the system in response to hazards that are believed to exist but do not actually exist. This is because false alarms are a form of false alert.

When it comes to the False Negative Rate (FNR), By comparing the overall number of malicious actions to the percentage of genuine malicious behavior that the intrusion detection system (IDS) is unable to identify, this indicator reflects the percentage of malicious activity that is actually occurring. This is made available by the following Equation:

$$FNR = \frac{False\ Negatives}{False\ Negatives\ +\ True\ Positives}$$

A high FNR can be much more harmful than a high FPR in many regards, particularly for SCADA systems that are responsible for key infrastructures. This is especially true in situations where the FNR is high. Each and every false negative indicates that there is a security problem that was not found, which might potentially lead to unauthorized access, data breaches, or even physical damage to components of the infrastructure.

A healthy equilibrium between FPR and FNR is of the utmost importance for SCADA systems, and it is essential that this equilibrium be maintained. The possibility exists that it would be extremely detrimental to either disregard a true threat (high FNR) or turn off a portion of a power grid or water treatment plant on the basis of a false warning (high FPR). Both of these actions might have a significant impact. This is due to the inherently significant character of the infrastructures that are involved in the situation. It is possible that either of these two categories of circumstances could result in disastrous consequences.

In order to conduct an accurate evaluation of intrusion detection systems (IDSs), it is essential to take into account both of these rates, as stated by the findings of a certain number of professionals. As an illustration, Anderson, Hall, and Bowyer (2016) conducted research on a variety of intrusion detection systems (IDS) in order to find the ideal balance between FPR and FNR. This was done in order to reach the optimal balance. This move, which was conducted in

order to assure both robust security and minimum disruptions in SCADA networks, insured that both of these things would be guaranteed.

2.6.3 ROC Curves and AUC Values

In the same way that other classification models can gain the benefits of complete performance metrics that reflect the intricacies of their detection capabilities, Intrusion Detection Systems (IDS) for SCADA networks can also reap the benefits of these sorts of measurements. The reason for this is because intrusion detection systems are able to recognize and eradicate invasions. In this particular setting, ROC curves and AUC values are of the utmost significance, and it is impossible to overstate the significance of these statistics. Furthermore, they offer a holistic perspective of the actual positive rate of a system in relation to the false positive rate of the system. This is in addition to the fact that they offer insights that single-point metrics would overlook.

2.6.3.1 ROC Curves:

The Receiver Operating Characteristic (ROC) curve is a graphical representation that illustrates the performance of a binary classification system as its discrimination threshold is varied. On this graph:

- The x-axis represents the False Positive Rate (FPR).
- The y-axis represents the True Positive Rate (TPR) or Recall.

The ROC curve, which stands for receiver operating characteristic, is better able to visualize the trade-offs between the bad effects of false positives and the advantages of actual positives as a result of this. it is generated by plotting the true positive rate (TPR) versus the false positive rate (FPR) at a number of different threshold values. One point in the upper left corner of the ROC

space would represent an intrusion detection system (IDS) that has a true positive rate (TPR) of 1 (identifying all real threats) and a false positive rate (FPR) of 0. This would indicate that the IDS recognizes all real threats. On the other hand, it is possible that the number of false alarms will increase in tandem with the sensitivity of an intrusion detection system (IDS), which will result in an increase in the number of actual positives that are identified.

2.6.3.2 AUC Values:

In this context, the term "AUC" refers to the "Area Under the ROC Curve." This method's objective is to quantify the total capacity of the model to differentiate between positive and negative classes. For this goal, the methodology is being used. If the area under the curve (AUC) number is 1, it means that the classification is perfect. On the other hand, if the AUC value is 0.5, it indicates that the model's ability to differentiate between the classes is comparable to that of trying to guess at random.

The AUC offers a variety of significant advantages, including the following:

1. Independence with regard to the standard at which judgments are made: Unlike other metrics, which are dependent on a particular threshold for classification, the area under the curve (AUC) evaluates the performance of the model over all of the available thresholds. This is in contrast to other metrics, which are based on a certain threshold.
2. The management of the imbalance: In situations where there is a class imbalance, such as when there are more benign network activities than hostile ones in IDS datasets, the area under the curve (AUC) continues to be a trustworthy indicator, whereas accuracy may be misleading. This is because the AUC indicates the proportion of network activities that are beneficial to the system.

Correct classifications can have significant effects on SCADA systems, so the ROC and AUC values are quite helpful in this situation. This specific rationale explains why these measures are so valuable. By means of an examination of the whole performance of an intrusion detection system (IDS) over all thresholds, stakeholders will be able to make decisions more informed about which models to install or further enhance. This will enable them to make decisions more likely to help their goals to be fulfilled.

Many research has been carried out to provide some light on the relevance of ROC and AUC in deciding the performance of an IDS. The efforts of these works have helped this discipline to grow. Using ROC curves and AUC values, Tan, Sher, and Chaudhary (2017) for example examined the performance of machine learning-based intrusion detection systems (IDS) for SCADA networks. This was carried out to ascertain the efficacy of various methods. They underlined during the implementation of these policies how strong they are in the face of a wide spectrum of danger environments.

Chapter 3: Stealthy Deception Attacks in SCADA Networks

3.1 Conceptual Framework

In the realm of cyber threats, stealthy deception attacks, sometimes referred to as SDAs, are a risk that is both extremely complex and constantly evolving. The availability and integrity of SCADA systems, which serve as standards for supervisory control and data gathering, are the targets of these attacks, which are carried out with the specific intention of compromising their availability. Maintaining continuous network activity and making use of the inherent trust that is entrenched in industrial control systems are two of the ways that SDAs are designed to successfully avoid discovery. This is consistent with SDAs, which are designed to avoid detection, in contrast to traditional cyberattacks, which typically rely on methods such as brute force or malware, both of which may be identified in a short amount of time. The execution of these attacks is extremely dangerous because of their ability to remain undetected for extended periods of time, which makes them extremely destructive. Criminals are able to drastically degrade the system's operational and physical components before they are discovered thanks to its functionality. The conceptual foundation for the generation of SDAs is the capacity of these entities to gently impact the processes that occur inside the system. To do this, it is customarily necessary to make adjustments to the data obtained from the sensors or the directives for the control system in order to guarantee that it continues to fall below the anticipated operational thresholds of the system. In order to overcome anomaly detection systems, fake data injection (FDI) attacks involve the covert modification of sensor readings, such as temperature or pressure measurements, in order to deceive

operators into making incorrect judgments while simultaneously avoiding detection by anomaly detection systems (Mo et al., 2014). Other examples of FDI attacks include the manipulation of pressure values and temperature readings. One illustration of this would be the alteration of the values of temperature or pressure. One of the most notorious examples of this tactic was the attack that was carried out in 2015 on the electrical grid of Ukraine. In that particular event, the offenders interfered with the status of circuit breakers in order to cause a chain reaction of power outages while simultaneously displaying normal readings to the operators (ENISA, 2022).

To provide a more adequate depiction of the conceptual framework, it is vital to take into consideration the historical development of SDAs. This is because the historical growth of SDAs is essential. According to Rid (2012), the Central Intelligence Agency (CIA) is suspected of causing damage to a Soviet gas pipeline in the 1980s by injecting a logic bomb into the control software using a logic bomb. As one of the earliest examples of stealth strikes that can be demonstrated to have taken place, this is especially noteworthy. Despite the fact that this attack resulted in a significant explosion, it was designed to give the impression that it was a technical fault rather than a purposeful act of sabotage. This was done in order to give the impression that. More recently, the Stuxnet attack that took place in 2010 demonstrated that it is feasible for SDAs to cause physical harm while being undetected for extended periods of time. This was a revelation that became public knowledge. Changing the speed of centrifuges was the method that Stuxnet was meant to use in order to target Iranian uranium enrichment facilities, as stated by Langner (2011). Despite the fact that the centrifuges continued to exhibit normal readings to the individuals in charge of the operation, this caused them to spin out of control.

The portion of SDAs that pertains to psychology is also a crucial component that contributes to the efficiency of these programs. Attackers take advantage of the faith that operators have in

automated systems by exploiting the "normalcy bias" that encourages operators to assume that data integrity is present even when there are tiny anomalies present. This allows attackers to take advantage of the faith that operators have in automated systems. According to a survey that was issued by IBM in 2023 (IBM X-Force, 2023), 68 percent of industrial control system operators were unable to detect stealthy attacks. This was due to the fact that the malicious adjustments fell within the projected functioning thresholds. This highlights the significance of not just putting in place technical safeguards but also giving behavioral training to operators in order to aid them in identifying and reacting appropriately to even the most subtle indications of compromise.

The main driving force behind SDAs has been their development mostly from the great networking of SCADA systems. This has been the case from the dawn of their existence. It was usual practice for SCADA systems to be disconnected from outside networks when they were originally developed. These technologies are often combined with enterprise information technology systems and the internet in today's environment, though. Though operational efficiency benefits from this interconnection, it has also resulted in the development of new attack routes open to use by malevolent actors. The situation as it is now could be dangerous. Consider the security breach that happened in the Oldsmar water treatment plant in 2021, for instance. This vulnerability could have happened since an instance of TeamViewer was let to connect to the internet. This specific occurrence made it feasible for the offenders to alter the chemical concentrations in the water supply from a distance (CISA, 2021).

study of the architectural layout.

3.2 Architectural Analysis

Stealthy deception attacks, sometimes known as SDAs, have architecture that is a multi-stage approach aimed to bypass the several safeguards set by SCADA networks. Also usually referred to as SDAs are SDAs. This lifetime consists mostly on reconnaissance, gradual intrusion, persistence, and payload execution. These are the elements regarded to be most important. Realizing that every one of these components is crucial in deciding whether or not the attack is successful is of great relevance. Referred known as reconnaissance, the first phase of the process is reconnaissance: obtaining information about the network under inquiry. This approach will involve compiling data about the network's architecture, protocols, and weaknesses. Potential attackers regularly utilize tools like Shodan, a search engine for devices linked to the internet. These technologies are meant to find insecure Programmable Logic Controllers (PLCs) or Human-Machine Intervals (HMIs) prone to hacking. This allows vulnerable devices prone to misuse to be found and recognized. The attackers of the Oldsmar water treatment system started their search for suitable TeamViewer instances online in 2021. Once they came across such examples, they made use of them to get system remote access (CISA, 2021). Here is one plausible method the breach might have started. Since it allows attackers to map the network and identify weak points so they may launch attacks, the discipline of reconnaissance is of great relevance. Two examples of these vulnerable sites are software not updated or incorrectly configured firewalls.

After the successful acquisition of access, the next action—which is frequently called incremental intrusion—occurs. This one defines the next action. By now the attackers will be using a range of techniques to get onto the network. Under this area are some of the techniques phishing, computer password theft, and zero-day vulnerabilities fit. One of the most well-known instances of this degree is the virus Stuxnet, which was spread in 2010 with the goal of undermining Iranian

nuclear facilities. This specific worm is a great illustration of the present mechanism in action. Langner (2011) claims that Stuxnet, a hostile programmable logic controller (PLC), could have entered the target network and brought about modifications. It managed this by concentrating on four zero-day Windows vulnerabilities and spreading them using USB drives. Stuxnet effectively achieved these goals. Using this sluggish approach—that which involves changing the system to seem to have little impact on reaching this goal— attackers can hide themselves and evade detection. In the case of Stuxnet, for instance, the malicious program progressively raised centrifuges' speed, which finally caused them to fail over time while the operators kept getting normal readings. This was the situation even though the operators kept getting normal readings. This was the reality even though the centrifuge was running as usual. Given the intricacy of SDAs, the detecting method makes great difficulty to recognize them. Among other things, this is what sets them apart from others.

The objective of the third stage, sometimes called as persistence, is to keep using the network while also evading discovery. The third stage aims at this. The achievement of this goal is reached by regularly applying a broad spectrum of techniques. Under this category there are several techniques like log manipulation, encrypted communication routes, and rootkits that fit. Based on the results of a survey Dragos conducted and made public in 2022, almost forty-five percent of all hacks that occurred in industrial control systems involved the alteration of logs in order to eradicate any signs of an incursion (Dragos, 2022). For example, the Triton virus, developed in 2017 meant to target Saudi petrochemical companies, used encrypted pathways to evade detection and simultaneously disabled safety mechanisms (FireEye, 2018). This was done to help to achieve both of these objectives. This stage of tenacity is very crucial since it helps attackers to keep under

control the system and accomplish their objectives without being found. This explains the great relevance of it.

Payload execution, the last phase, is the obligation of the attacker to engage in the damaging behavior meant for them to perform. The onslaught reaches its last stage here. One could fall under this category when changing sensor data or control directives. By demonstrating that they could progressively raise chlorine levels by 15% without triggering alarms in a simulated attack on a water treatment plant, researchers were able to show that SDAs had the capacity to cause major damage without being noticed (Adepu & Mathur, 2016). This was achieved by proving that they could raise chlorine levels by 15% without setting off alarms. For instance, this shows how scientists were able to show that SDAs could cause significant damage undetectable to humans. Regarding the effective payload execution, this phase is sometimes the most disastrous. This is so because it can physically damage infrastructure, disturb running operations, and even result in fatalities. This is the reason one should exercise great care.

The architecture of SDAs emphasizes the systematic nature of the shortcomings in SCADA security. There are architectural flaws including these ones. Among these flaws are outdated firmware and a too reliance on air-gapped networks. Both of these problems exist. The results of a SANS Institute poll conducted in 2023 (SANS, 2023) show that sixty-two percent of operators of critical infrastructure still use Windows XP or earlier versions of the operating system, which lack fixes relevant to vulnerabilities that have lately been discovered. SCADA systems are particularly vulnerable to attacks carried out by hostile software since they depend on outdated systems. Attackers can take use of found but unreported flaws in systems. These weaknesses never have been fixed. This explains the state of affairs as they are.

Considering the tools and strategies attackers use across the several phases of the attack lifetime helps one to present a more complete picture of the architecture study. This is so since attackers make use of them during the attack life. To reach this objective, one must evaluate the instruments and strategies attackers use. For example, attackers regularly employ open-source intelligence (OSINT) technologies as Shodan, Censys, and Nmap in the reconnaissance phase to find devices that might be targeted. This is carried out in search of equipment prone to attack. This is meant to help find and mark devices that might be targets of attack. Attackers could search the internet for devices running certain protocols, such Modbus or DNP3, which are often used in SCADA systems. By means of these technologies, they can hunt for devices operating under specified criteria. Once a vulnerable device has been found, attackers can enter the system by using exploit frameworks like Metasploit. The system is vulnerable to assault; hence this is doable. This is within the range of possible as soon as the gadget is located.

During the phase of incremental penetration, attackers often utilize phishing emails and other kinds of social engineering to get access to the security of the network. This is done to be on the network. Acting in this way helps one to get network access. For example, fraudsters used spear-phishing emails to entice employees into downloading harmful software (ENISA, 2022) in the attack conducted on Ukraine's 2015 energy system. Cybercrime carried out this attack. By means of lateral movement strategies, attackers can migrate from one device to another, progressively extending their rights until they can seize command of critical systems. This lets attackers raise their access degree gradually. Once they acquire network access, they can use these strategies to go laterally from one device to another. They can so move from one gadget to another. Usually, the persistence phase consists in the application of more complex techniques, such rootkits or bootkits, to keep authorization to access the system. This is so true as the persistence phase is

focused on maintaining system access. For instance, the Havex malware aimed targeting energy businesses in both Europe and North America using a rootkit to hide its presence on systems that were infected (Symantec, 2014). This infection was meant to go after energy firms in both areas. This hostile program aimed at both geographical areas, meant to attack energy companies. Two goals attackers can reach using encrypted communication channels include filtering data or getting directives from a command-and- control (C2) server. Another way attackers might achieve their objectives is this one. Finally, but most definitely not least, the stage of payload execution sometimes entails the use of malicious software meant specially to control particular systems or processes. One example of this would be the malevolent program called Industroyer, created to control circuit breakers (ESET, 2017) and cause general power outages. Built specifically with the goal of attacking Ukraine's electrical system, this malicious program emerged in 2016. This level of complexity makes it extremely crucial to have a multi-layered security plan including human-centric, procedural, technical in character elements. This is carried out to offer defense against possible attacks.

3.3 Examples and Real-Life Case Studies

Many well-publicized incidents utilizing stealthy deception attacks—also known as SDAs—show their capacity to substantially undermine operational and physical infrastructure. A successful attack on vital infrastructure such water treatment plants, oil refineries, and electricity grids can have disastrous repercussions. Often these strikes target key infrastructure. Among the most well-known cases of a sophisticated distributed assault (SDA), the worm known as Stuxnet concentrated on Iranian uranium enrichment facilities in 2010. The Stuxnet computer program sought to change the speed of centrifuges, hence pushing them to spin out of control and concurrently show operators regular data. This attack highlighted how physically harmful SDAs may be even while

they are unnoticed for long spans of time (Langner, 2011). The sophistication of Stuxnet, which utilized new techniques of eluding detection and used several zero-day vulnerabilities, marked a turning point in the development of cyber threats to industrial control systems (ICS). Stuxnet exploited once unknown many flaws.

The 2021 hack at the Oldsmar water treatment plant is another noteworthy SDA. In this incident the attackers remotely altered the sodium hydroxide concentration in the water supply using a hijacked TeamViewer instance. Emphasizing the difficulties in real-time SDAs identification (CISA, 2021), the assault was discovered only when an operator noticed some strange cursor movement on the HMI. This occurrence revealed the shortcomings in remote access systems as well as the requirement of protecting equipment connected to the internet in important infrastructure. The Oldsmar attack also revealed the possibilities for SDAs to create public health concerns since the change of chemical levels in drinking water may have had disastrous results if it had not been found in time. The attack helped to demonstrate both of these possible outcomes.

The Triton malware, which first surfaced in 2017 and targeted Saudi petrochemical businesses then, is another illustration of the complexity of SDAs. Triton was created to sabotage safety systems, therefore compromising the probability of explosions and other catastrophic incidents. FireEye (2018) says secret communication paths permitted the damaging software to hide itself while it was altering safety settings. This attack made it abundantly evident that SDAs might target safety systems, which are absolutely vital in industrial situations to prevent accidents. Especially concerning was malware known as Triton as it demonstrated attackers' ability to influence security systems. In manufacturing sites, safety systems are usually seen as the last line of protection.

These incidents clearly demonstrate that effective detection and response policies are required since SDAs can have disastrous outcomes. According to the Cybersecurity and Infrastructure

Security Agency (CISA, 2023), 72% of US water utilities lack intrusion detection systems able of identifying secretly carried out attacks. This material is crucial since it underlines the need of increasing expenses in cybersecurity policies and the shortcomings of major infrastructure to SDAs.

One should offer a more whole picture of the effects of SDAs by including other case studies and actual happenings. For example, the use of spear-phishing emails provided access to the network, therefore facilitating the attack on the Ukrainian power grid in 2015 that resulted in notable disruptions of the electricity (ENISA, 2022). Once on the network, the attackers changed circuit breakers using malware to cause cascade failures.This attack underscored the need of protecting industrial control systems (ICS) networks from phishing and other kinds of social engineering at the same time it also showed the possibility for SDAs to cause huge disruptions.

Another example is the spyware known as Havex, which targets North American and European energy businesses. Designed mostly for intelligence collecting on industrial control systems and passing that data to a command-and- control (C2) server, according to Symantec (2014), the hostile program hides its presence on compromised systems using a rootkit, therefore complicating detection efforts. Given this attack, it is rather clear that SDAs might be exploited for espionage. This is so because attackers might gain important information about industrial operations and then apply that knowledge to plan next targets.

Another instance of the complexity of SDAs is the 2016-introduced virus known as Industroyer, which targeted Ukraine's electrical system. Industroyer was expected to be designed to answer for major power outages and control circuit breakers. According to ESET (2017), innovative methods were utilized to elude detection and the malicious program was able to seriously compromise the

electrical infrastructure. This attack underscored the need of using robust detection and response systems in important infrastructure since it proved that SDAs might cause major disruptions.

Apart from the well-publicized instances depicted above, SDAs have concentrated on critical infrastructure in a great number of other cases. For instance, in 2019 Massachusetts Institute of Technology (MIT) researchers showed that SDAs might regulate sensor data in gas pipelines, therefore producing unnoticed breaches with possible explosive consequences. This investigation highlighted the susceptibilities of sensor networks to assaults involving fraudulent data insertion (FDI) as well as the prospect that SDAs could physically harm infrastructure.

2020 saw an attack on a German hospital running ransomware. Operations were hampered and the facility was forced to shut down significant systems by ransomware. Though this attack was not sponsored by any state (SDA), it does highlight how seriously cyberattacks could impair vital infrastructure (BBC, 2020). This episode underscored the need of implementing rigorous cybersecurity rules for significant infrastructure, which should include the ability to identify and manage covertly carried out attacks.

Apart from the effects on property and disruptions of business activities, SDAs can cause damage to other aspects. Moreover, these strikes could have very important financial implications. One example of this would be the 2017 NotPetya ransomware epidemic that aimed businesses all around the globe. According to Greenberg (2018), attack damage came to total about $10 billion. Although NotPetya was not a state-sponsored breach, it revealed the great financial damage that capable hackers could inflict.

Growing threat from SDAs has caused governments and businesses to pay more attention. For instance, the Department of Energy (DOE) of the United States of America has created

recommendations including SDAs (DOE, 2021) to help to protect vital infrastructure against cyber threats. These points underscore the need of developing multi-layered protection systems. Among these approaches is network segmentation, intrusion detection systems, and regular security audits.

Apart from government projects, business groups have also responded to handle the problem given by SDAs. For instance, the International Society of Automation (ISA) created policies aimed to safeguard industrial control systems. These criteria correspond with the ISA/IEC 62443 series (ISA, 2020). The operators of critical infrastructure make great use of these rules since they offer a framework for the implementation of cybersecurity rules in industrial surroundings. Although these steps have been done, SDAs pose increasing risk. From 18% in 2018 to 32% of all Industrial Control System (ICS) issues are under the accountability of SDAs; this is a rise from 18% in 2018. This evolution highlights the significance of maintaining investments in cybersecurity measures as well as the increasing degree of cybercrime related skill level.

3.4 Sensitivity Points and Attack Scenarios

In addition to being referred to as SDAs, surreptitious deception assaults are attacks that make use of specific vulnerabilities that are present in SCADA systems. These flaws can let hackers gain access to the system. In most cases, the targets of these attacks are identified as the essential components that are necessary for the effective and safe operation of industrial operations. Control loops, communication protocols, sensor networks, and human-machine interfaces (also known as HMIs) are all examples of weaknesses that are referred to as sensitivity points. Sensitivity points are also examples of vulnerabilities. There is another name for these vulnerabilities, and that is the term "sensitive points. With a comprehensive awareness of these sensitive areas, it is possible to anticipate possible attack scenarios and devise strategies to mitigate

the risks posed by SDAs. This is a possibility that can be considered. The reason for this is that both of these things are indeed feasible.

1. Sensor Networks

The primary focus of sensor data acquisition systems, also known as SDAs, is on sensor networks. This is due to the fact that these networks provide the data that operators rely on in order to create decisions. Sometimes referred to as FDI attacks, fake data injection attacks are a frequent strategy used by attackers to fool operators and disrupt the processes being followed. These attacks consist on small changes to the readings that sensors produce. For example, 2019 Lincoln Lab research at the Massachusetts Institute of Technology found that a 5% data skew in pressure sensors can possibly postpone the leak detection in gas pipelines by forty-48 hours (MIT, 2019). This delay increases the possibility of major physical damage including explosions and environmental consumption by harmful chemicals.

Because many industrial protocols, including Modbus and DNP3, lack authentication and encryption capability, sensor networks are much more vulnerable to assaults from foreign direct investment (FDI). Built with dependability in mind rather than security in mind, these systems are easily manipulated by attackers (NIST, 2020). This is thus since the building of the protocols considered dependability. For a simulated attack on a water treatment facility, for instance, researchers were able to progressively raise chlorine levels by fifteen percent without triggering alarms. Adepu and Mathur (2016) claim this shows how seriously attacks on foreign direct investment can damage without being acknowledged. Establishing strong data validation techniques—such as algorithms for anomaly detection and cross-verification of sensor data—that would help to lower the dangers presented by assaults involving foreign direct investment is absolutely necessary. One may use machine learning techniques, for instance, to identify trends

in sensor data suggesting hazards to foreign direct investment (Yin et al., 2020). This is a doable thing. Furthermore, the use of safe connection techniques including Transport Layer Security (TLS) can help to prevent hostile actors from intercepting or altering the data obtained by sensors (IBM X-Force, 2023). This help can be quite helpful in ensuring the data stays unaltered or stolen.

2. Control Loops

An additional essential sensitivity point is provided by the control loops in SCADA systems. These control loops are accountable for the regulation of operations such as temperature, pressure, and amount of flow. Control loops are responsible for regulating the processes, which is why this is the case. They are responsible for regulating activities, and this function is the one that is accountable for these loops. Because these control loops are dependent on the output received from sensors in order to adjust control parameters, they are subject to manipulation by adversaries. This is because the output is collected in order to modify control settings. Because of this, individuals are especially vulnerable to being deceived compared to other people. For instance, researchers proved that modifying the setpoints of proportional-integral-derivative (PID) controllers can result in frequency changes that cause generators to trip, which ultimately leads to widespread power outages (ENTSO-E, 2022). This was verified by the researchers. There are many ways in which something like this can occur. There was a simulated attack on the energy system of Europe that took place in the year 2022. This was done as a means of demonstrating that this allegation is true.

Because of the presence of ancient systems and software that has become obsolete, the sensitivity of control loops to SDAs is further increased. This is because of the fact that SDAs are a source of sensitivity. This makes the problem even more challenging to handle than it already was. According to the findings of a survey that was carried out by the SANS Institute in 2023

(SANS, 2023), it was discovered that sixty-two percent of operators of critical infrastructure continue to use Windows XP or earlier versions of the operating system. These versions of the operating system do not have fixes that are applicable to vulnerabilities that have been discovered more recently. These outdated systems are prone to be used by hostile actors since they usually lack the capacity to include modern security methods such encryption and authentication. This makes them easily targets for exploitation. This makes them susceptible to exploitation as a result. Though there are many of services of this nature, two that really stand out as very important are authentication and encryption.

Regarding SDAs, it is rather crucial to provide strong access control systems including multi-factor authentication (MFA) and role-based access control (RBAC) to safeguard control loops across the application implementation. This is so since these systems are necessary to guarantee that only authorised people can access them. This is justified by the great importance of ensuring that control loops are not in any form compromised. This makes the development of these systems of great relevance in respect to the current state. The use of intrusion detection systems (IDS) can show to be helpful in the search and reaction to illegal changes in control settings (Dragos, 2022). It distinguishes it since this is an extra point of interest. By means of algorithms that identify anomalies in order to find deviations from expected patterns, one can monitor the behavior of control loops and thereby uncover deviations from projected patterns. This is a reasonable method. This is something one can undertake to identify trends different from those projected earlier. Here is an example of one that could be carried out in a particular way for illustration. These differences could be a sign of a notable deviation from those that are usual, as the results of the 2020 research conducted by Khraisat and associates point to.

3. Communication Protocols

Regarding SCADA systems, one of the most absolutely important sensitivity areas is still communication channels. Among the systems falling under this category are Modbus, DNP3, and OPC (Open Platform Communications), to mention a few. These protocols are a great target for smart device applications (SDAs) since they are used for the data transmission between HMIs, controllers, and sensors. This is the case considering that these systems are used for data transfer. Moreover, according to ENISA (2022), malware was used by those in charge of attacking the Ukrainian electrical grid in 2015 to distort Modbus transmission and generate failures that cascaded. As things were during the assault.

The fact that many industrial protocols lack encryption and authentication mechanisms considerably increases the vulnerability of communication systems to SDAs. Using the SCADA system Modbus as an example, it is clearly vulnerable to the prospect of being intercepted and under control by intruders since it lacks security precautions included in it (NIST, 2020). Modbus is much used in SCADA systems. Furthermore, the use of non-encrypted communication channels like TCP/IP could expose SCADA systems to risk of being targets for assaults including a man in the middle. IBM X-Force (2023) claims that these attacks are those in which the offenders intercept and control data as it is in transit by means of a computer network. Should SCADA systems be applied, they could be vulnerable to these attacks.

Using safe communication techniques is absolutely crucial when it comes to lowering the risks that SDAs aiming at communication protocols create. Two such examples of the several approaches that fit under this category are VPNs and transport layer security (TLS). Moreover, the use of security extensions tailored for protocols—like Modbus Secure and DNP3 Secure—may help to guard industrial systems against attack by enemies (FireEye, 2018). These extensions

are meant to prevent the protocol tampering. For instance, Modbus Secure features encryption and message authentication, which help to guard SCADA systems from being accessed by anybody not authorized to do so (ISA, 2020).

4. Human-Machine Interfaces (HMIs)

The human-machine interfaces (HMIs) found in SCADA systems offer even another quite important sensitivity point. Human-machine interfaces (HMIs) enable operators to monitor and run systems in real time by giving them a graphical representation of the several related processes in industrial operations. Conversely, human-machine interfaces (HMIs) depend on data from controllers and sensors, which can be controlled by enemies, hence they are usually vulnerable to particular data attacks (SDAs). For instance, the attackers exploited a TeamViewer instance hijacked in order to remotely change the chemical levels in the water supply in the water treatment breach that occurred in Oldsmar in 2021. This was undertaken to regulate the water supply. These changes were presented on the HMI in line with industry standards (CISA, 2021).

An increase in the sensitivity of human machine interfaces (HMIs) to safety data acquisition (SDAs) corresponds with the use of remote access technologies, such TeamViewer and VNC (Virtual Network Computing), which are extensively used to provide operators remote access to SCADA systems. Commonly configured with passwords that are not very strong and devoid of multi-factor authentication (MFA), these systems expose themselves to the possibility of being used by hostile actors (IBM X-Force, 2023). Furthermore, lacking multi-factor authentication are these systems. Moreover, the use of outdated software and firmware in human-machine interfaces (HMIs) could expose SCADA systems to known flaws that attackers could take advantage of to get illegal access (SANS, 2023). This could endanger a business that wants to keep its security.

Establishing strong access control methods such multi-factor authentication and role-based access control can help to defend human machine interfaces (HMIs) from security device attacks (SDAs). Two safe remote access options that would be quite helpful in stopping unwanted access to SCADA systems (Dragos, 2022) are VPNs and zero-trust networks. These fixes allow one to stop illegal access to SCADA systems. Regarding zero-trust networks, for instance, users must personally validate themselves and grant their consent for every single access request. This helps to lower the likelihood that illegal access to human-machine interactions (NIST, 2020) might occur.

5. Attack Scenarios

One can locate prospective attack scenarios after one understands the sensitive areas present in SCADA systems. Following these scenarios allows one to create and apply strategies of action that serve to lower the risks presented by SDAs. For instance, the following lists typical assault techniques:

1. In water treatment facilities, one can adjust pH by means of Attackers could operate pH sensors to gradually increase or decrease the pH level in drinking water, hence generating corrosion or conditions harmful for public health. The general health of the people can suffer as thus. For a simulated attack on a water treatment plant, for example, researchers were able to steadily elevate chlorine levels by 15% without setting off alarms (Adepu & Mathur, 2016). This shown how effectively the researchers achieved their goal. Gradually increasing the chlorine content of the fluid will help one reach this.
2. Slow and constant. In power systems, voltage-adjusting switches Attackers could be able to control voltage setpoints in power networks, hence producing slow changes that overload transformers and cause large blackouts. The security and safety of the United

States of America could be compromised here. Changing setpoints in PID controllers might, for example, generate frequency fluctuations that might trip generators during a simulated attack on the European power grid in the year 2022 (ENTSO-E, 2022). We repeated this attack using 2022.

3. Attackers can change the pressure sensors in invisible gas pipelines, therefore causing undetectable breaches that could lead to explosions. This has resulted in leaks in gas pipelines not recorded causing explosions. For instance, a 2019 Lincoln Lab study at the Massachusetts Institute of Technology revealed that a 5% data skew in pressure sensors might significantly delay leak detection by one day (MIT, 2019).
4. 4. Turning off safety systems in processing petrochemical factories: Attacks can target petrochemical sectors since they can modify safety mechanisms to turn off warnings and generate explosions. For example, the Triton virus, intended in 2017 to disable safety systems and increase the risk of explosions, was largely aimed on Saudi petrochemical enterprises (FireEye, 2018).

These attack scenarios clarify the probable results of SDAs as well as the necessity of creating robust defenses for both reaction and detection. Comprehensive awareness of the sensitivity points in SCADA systems as well as the numerous attack scenarios assists operators of critical infrastructure to better defend their systems and lower the hazards offered by SDAs. This helps them to fortify the protection of their systems. Textual representation of a table summarizing the key incidents and statistics related to Stealthy Deception Attacks (SDAs). Table 1 provides a review of the largest cyber-attacks against industrial control systems (ICS) and SCADA systems in the last two decades, which depict the victims, attack patterns, and the outcomes. These are the Stuxnet, Triton, and Industroyer attacks, and they show how severe the attack on the critical

infrastructure like power and water grids and industrial plants can be, and the effects can be both physical damage and extensive disruption of various services and endangered lives of people. According to the latest news, the number and complexity of such attacks has been on the rise, and more is to be done in order to enhance intrusion detection and industrial cybersecurity infrastructure to safeguard important plants.

Table 1. Major Cyber Incidents Targeting Industrial Control and SCADA Systems

Incident Name	Year	Target	Attack Method	Consequences	Statistics/Notes
Stuxnet	2010	Iranian uranium enrichment facilities	Manipulation of centrifuge speeds	Physical damage to centrifuges; prolonged undetected operation	Exploited multiple zero-day vulnerabilities; marked a turning point in cyber threats.
Oldsmar Water Treatment Breach	2021	Oldsmar, Florida water treatment	Remote access via compromised TeamViewer	Risk of public health crisis due to chemical manipulation	Detected by operator noticing unusual cursor movements; highlighted remote access vulnerabilities.
Triton Malware	2017	Saudi petrochemical plants	Disabling safety systems	Risk of explosions and catastrophic events	Used encrypted communication to evade detection; targeted critical safety systems.
Ukrainian Power Grid Attack	2015	Ukraine power grid	Spear-phishing emails leading to malware deployment	Widespread power outages	Demonstrated potential for cascading failures; emphasized need for securing ICS networks.
Havex Malware	2014	Energy companies in Europe/North America	Intelligence gathering on ICS	Data exfiltration; potential for future attacks	Used a rootkit to hide presence; highlighted espionage potential of SDAs.
Industroyer	2016	Ukraine power grid	Manipulation of circuit breakers	Significant damage to power grid; widespread outages	Advanced evasion techniques; targeted critical infrastructure.
Gas Pipeline Sensor Manipulation	2019	Gas pipelines	False data injection	Undetected leaks leading to potential explosions	Highlighted vulnerabilities of sensor networks to SDAs.
German Hospital Ransomware Attack	2020	German hospital	Ransomware disruption	Forced shutdown of critical systems	Demonstrated potential for cyberattacks to disrupt critical infrastructure.
NotPetya	2017	Global businesses	Ransomware attack	Estimated $10 billion in damages	Not an SDA, but demonstrated economic disruption potential of cyberattacks.
CISA Report	2023	U.S. water utilities	Lack of intrusion detection systems	78% of water utilities lack adequate detection systems	Highlights vulnerabilities in critical infrastructure to SDAs.
Mandiant Report	2023	Industrial Control Systems (ICS)	Various SDA incidents	SDAs account for 32% of ICS incidents, up from 18% in 2018	Indicates growing sophistication of attackers; need for continued cybersecurity investment.

Chapter 4: Machine Learning in SCADA system

4.1 Introduction to Machine Learning in SCADA Security

It has lately been clear that machine learning (ML) is a technology with great potential to revolutionize the world of cybersecurity by providing original methods to recognize and react to attacks in real time. This is something that lately become clear. Supervisory Control and Data Acquisition (SCADA) systems depend much on machine learning approaches. This is so because these methods can examine enormous amounts of data, spot trends, and point up anomalies suggesting a cybercrime. Cyber threats more advanced are progressively focusing on SCADA systems, which monitor and control important infrastructure including power grids, water treatment plants, and oil refineries. One instance of this is the growingly popular sneaky deception assaults (SDAs). SCADA systems also find use in monitoring and controlling vital infrastructure. Conventional intrusion detection systems (IDS) often find it difficult to spot these sophisticated risks. This is thus because these systems depend on set rules and signatures. Attackers utilize subtle and always changing strategies, hence these rules and signatures might not be able to explain them. Machine learning offers a hopeful solution to this challenge since it can learn from data and adjust to new hazards. One finds this answer in machine learning.

The particular features of SCADA systems, which inspire the choice, encouraged the introduction of machine learning into control and data acquisition (SCADA) security. From sensors, controllers, and other devices, SCADA systems produce enormous volumes of time-series data to provide a rich source of information for machine learning algorithms to probe. One can examine the mechanism of machine learning using this information. A water treatment facility

can produce among other kinds of information data on pH, temperature, and flow rate. Conversely, a power grid can generate information on frequency, voltage, and current. Apart from that, a power grid might generate data on flow rate. Analyzing this data helps machine learning algorithms to spot anomalies that might point to a cyberattack and to find trends typical of ordinary operation. One achieves this by means of data evaluation. This talent is especially important since it enables one to identify SDAs, which are meant to evade already in existence detection systems by modeling moral behavior.

Figure 6 is a representation of a comprehensive machine learning framework that was designed for Intrusion Detection Systems (IDS) in SCADA (Supervisory Control and Data Acquisition) networks. This framework was developed for the purpose of managing machine learning. Initially, the primary focus is on the collecting of data from SCADA network traffic, system logs, and process parameters. This continues until the process is completed. With this data, a number of preprocessing techniques, such as feature extraction, time-window alignment, and class balancing, are carried out. Following this, an optimization feedback system is put into place in order to enhance the overall performance of the data. The framework's model architecture is that which is considered to be the most crucial component of the framework. A wide range of techniques, including decision trees, support vector machine classifiers, deep neural networks, random forests, and ensemble voting, are utilized in its operation. In the process of analyzing these models, a number of different measurements are applied. Some of these measures include the confusion matrix, ROC analysis, and adversarial testing. Eventually, the integration of the intrusion detection system (IDS) into the monitoring and control (SCADA) network is required for deployment, which is the final step. Real-time monitoring, alarm systems, and response protocols are all components of deployment. The importance of this structured approach in assuring the

dependability and adaptability of intrusion detection systems (IDS) for industrial control systems has been brought to light by research that was recently carried out on intelligent SCADA cybersecurity.

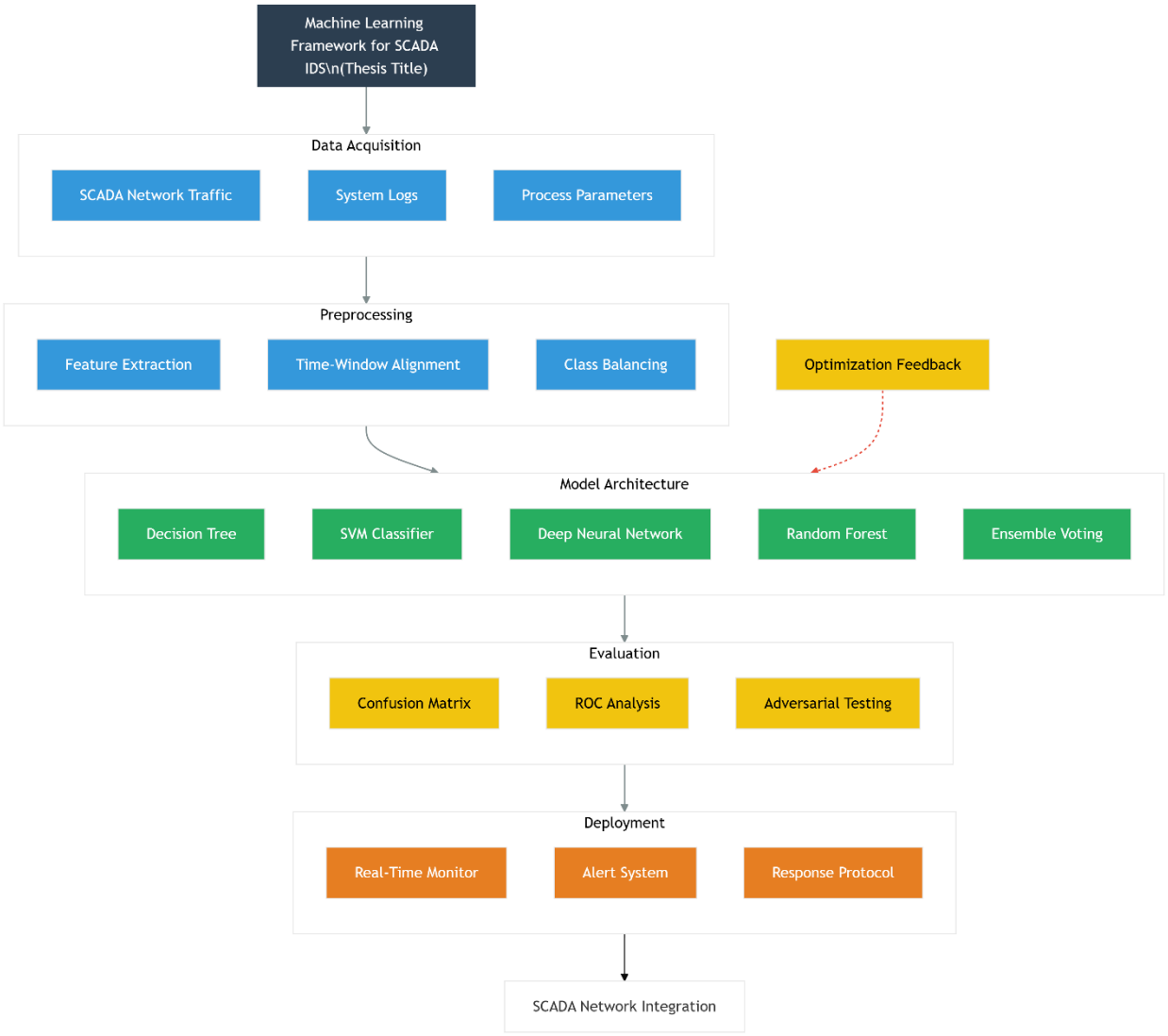


Figure 6 Framework Diagram

One of the most essential of the several great advantages of this technological development is the possibility of machine learning in SCADA security to handle the complexity and diversity of industrial environments. SCADA systems are mostly meant to run in conditions that are not only dynamic but also varied. This suggests that elements including the wear and tear on the

equipment, the state of the surroundings, and the changes made to the procedures could cause the behavior of the system to evolve with time. Traditional intrusion detection systems (IDS) may find it difficult to adapt to the changes taking place since they are based on stationary rules and signatures. One could expect false positives and false negatives from this situation. Conversely, by means of constant learning from new data, machine learning may adapt to changes in the system. This lets it fit more challenging circumstances. This helps one to better spot fresh dangers as they arise in the surroundings.

This technology also has the ability to identify hazards that were not previously acknowledged. Machine learning provides this capability. While conventional intrusion detection systems rely on known attack signatures, it is conceivable that these signatures are useless against constantly changing or new attacks. Conversely, even if the particular details of the assault are not yet known, machine learning can detect anomalies in the data that would indicate a new kind of attack. This is thus because machine learning may identify data irregularities. In the framework of SCADA systems, where the results of a successful assault could be disastrous, this capacity is quite crucial since it has the capacity to have disastrous results. A cyberattack on a power grid would be one instance of this, maybe causing significant power disruptions. Conversely, should an attack be directed against a water treatment plant, the safety of drinking water could be compromised.

Notwithstanding these benefits, using machine learning in SCADA security causes certain difficulties that have to be resolved. Conversely, one of the most important obstacles that has to be solved is the necessity for highly quality training data. To learn the patterns of normal and harmful behavior, machine learning algorithms depend on very massive volumes of tagged data. This is so as the algorithms must grasp behavioral patterns. Getting tagged data in the framework of SCADA systems can be challenging given the sensitive character of the data and limited

availability of real-world attack data. This helps one to realize that the work could get more challenging. Moreover, the noisy and fragmented data produced by SCADA systems might make learning information in an efficient way challenging for machine learning algorithms. For the algorithms, this could present difficulties.

Demand for real-time detection presents still another obstacle that has to be overcome. Since SCADA systems run in real time, any delay in identifying and handling a cybercrime could have major consequences. This is thus because real-time operation of SCADA systems is possible. Particularly those depending on sophisticated models like deep neural networks, machine learning techniques may use a lot of computing resources and time to process data. This is particularly true with methods depending on deep neural networks. This would make real-time detection increasingly difficult to reach. Overcoming this difficulty, scientists have created lightweight machine learning algorithms and edge computing systems able to undertake local data processing and offer real-time detection powers. These answers are meant to satisfy the difficulty.

Furthermore, raised questions are about the interpretability and openness of machine learning models resulting from their use into SCADA security systems. Because humans cannot adequately understand the results of many machine learning algorithms—especially deep learning models—many of them are commonly referred to as "black boxes". This is thus because human understanding of the decision-making mechanisms of various algorithms is challenging. Given the lack of openness resulting from this circumstance, operators may find it challenging to understand and have faith in the conclusions taken by machine learning-based identification systems. To solve this issue, experts in artificial intelligence have created techniques known as explainable artificial intelligence (XAI). By means of these approaches, which offer insights into

such processes, the decision-making procedures of machine learning models become more open and understandable.

Using machine learning in SCADA security calls for moral and legal issues that must be considered. These elements exist in addition to the listed technological challenges. For example, the use of machine learning techniques for monitoring and controlling significant infrastructure could lead to issues about responsibility and guilt in the case of a cyberattack. The gathering and use of data for the aim of training machine learning algorithms might lead to other privacy problems. This is particularly relevant in relation to SCADA systems, which oversee important infrastructure.

Notwithstanding these difficulties, machine learning in remote control and data acquisition (SCADA) security seems to have quite great promise. Machine learning allows companies to improve their capacity to identify and react to cyber-attacks, therefore lowering the likelihood of both physical damage and operational interruptions. Application of machine learning makes this feasible. Using machine learning-based intrusion detection systems (IDS) to find anomalies in sensor data that would be indicative of a fake data injection (FDI) assault would be one instance of this. This gives operators the chance to lessening the effect of the strike before it does major harm. In the same vein, network traffic can be examined using machine learning techniques to identify trends suggesting the existence of a cybercrime. One can accomplish this in a like manner. These trends could contain unusual communication patterns or attempts at access without authorization that go against convention.

Machine learning is still in its early phases in the field of SCADA security, hence there is great potential for future improvement in this specific discipline. Researchers are investigating the potential of enhancing the capabilities of intrusion detection systems (IDS) in SCADA systems by

means of advanced machine learning approaches including reinforcement learning and federated learning, for instance. Scholars are thinking about this. One method that might be used to create adaptive intrusion detection systems (IDS) able to learn from their surroundings and finally improve their effectiveness over time is reinforcement learning. One can achieve this with this technology. Machine learning algorithms are trained by means of a trial-and-error procedure guided by this approach. Using federated learning—training machine learning models on distributed data sources—one can build intrusion detection systems (IDS) able to operate in distributed SCADA systems without violating data privacy. Federated learning helps one to do this.

4.2 Model Evaluation and Selection

Good detection of breaches in SCADA systems requires on adequate models being properly chosen and investigated in order for machine learning (ML). This fixes this. This makes the present condition as it is unambiguous. Many different machine learning techniques allow one to investigate enormous amounts of created time-series data from SCADA systems. This helps SCADA systems to detect anomalies implying a cybercrime as well as patterns implying constant operation. One could deduce the kind of the cybercrime from the seen patterns. One cannot exaggerate the need of underlining the fact that not all machine learning techniques are equally suitable for the particular difficulties presented by SCADA systems. Among the extra obstacles to be surmounted are the necessity for real-time detection, noisy and inadequate data handling, and the ability to spot covert deception attacks (SDAs). The most widely used machine learning models for SCADA intrusion detection are examined in this part together with their pros and drawbacks and suggestions on which model would be appropriate for a given range of applications.

Systems of supervised learning

Applied for intrusion detection, supervised learning models represent among the most often used techniques of data mining in SCADA systems. Every data point in these models has a known outcome since they learn on labeled datasets—for example, either normal or evil. Once the model is trained, depending on the known patterns it could locate new data points. Decision trees, support vector machines (SVMs), and neural networks are only a few of the several supervised learning techniques often utilized in SCADA security.

Simple but rather effective approach of supervised learning, decision trees divide data into branches based on the values of specified variables. Simple decision trees are actually rather effective tools for prediction. One of the reasons behind their especially efficiency for SCADA intrusion detection is interpretability. This suggests that operators could act fast. For this among other things they are fairly successful. For example, based on packet size or transmission frequency, a decision tree can label network traffic as harmful should it exceed a predefined level. Once this level is reached, the traffic is said to be carcinogenic. Still, decision trees overfit readily—especially in SCADA systems where trained on noisy or inadequate data—a regular occurrence—Khraisat et al., 2020 noted. This is especially true in circumstances when the data has been noisy or lacking. This particularly holds true in situations when the data has been insufficient or noisy. Usually, among other ensemble techniques, gradient boosting and random forests help to lower the effects of this difficulty. 2020 comes yin and associates. Research on FDI attack detection in SCADA systems find good degree of accuracy in random forests—a mix of various decision trees. Considering as a combination of random forests and decision trees, Support Vector Matrix (SVMs) Still another usually used type of supervised learning model accessible for SCADA intrusion detection are support vector machines, or SVMs. By effectively defining the hyperplane most likely separating the data points into different categories, the support

vector machines (SVMs) help them to run as they should. Apart from high-dimensional data, they can also manage non-linear correlations in network traffic records by use of kernel functions. Especially they assist with high-dimensional data. By means of packet size, periodicity, and IP address of the destination, an SVM could be used to classify network traffic as either regular or hostile. The characteristics of the network traffic could direct this categorizing. Large computationally heavy support vector machines (SVMs) challenge real-time identification (NIST, 2020). This is especially true considering enormous volumes of data.

The ability of neural networks to identify complicated data patterns is one of the main factors behind their increasing popularity during the past several years. More precisely, deep learning techniques allow one detect these trends. Time-series data evaluation of sensors and controllers helps neural networks used in SCADA systems explore and discover anomalies. One can contribute to this using neural networks. Using a recurrent neural network (RNN) to investigate trends in sensor data over a period of time would allow one to identify foreign direct investment (FDI) difficulties, for example (Kim & Kang, 2019). This would assist to identify any risks to FDI negotiations. On the other hand, for training neural networks need a lot of labeled data, which SCADA systems could find somewhat difficult. Since neural networks learn from their own data, this is thus also true. These "black-box" models reportedly render IBM X-force (2023) less interpretable than support vector machines (SVMs.) or decision trees. This is true as these models vary really greatly. This could be a disadvantage in terms of the requirements for the running of essential infrastructure and in which openness is quite important.

Unsupervised Learning Models

The use of unsupervised learning models is employed in circumstances in which labeled data is either not accessible or is insufficient. The data that has not been tagged is subjected to an analysis

by these models in order to identify patterns or clusters that may indicate the presence of anomalies. Unsupervised learning models that are applied in SCADA security include clustering algorithms, autoencoders, and one-class support vector machines, to name a few examples of notable models.

Clustering techniques such DBSCAN and k-means group data points depending on individual point degree of similarity. SCADA systems can employ clustering methods to find unexpected trends in sensor data that might indicate a cyberattack aimed against the system. This would cluster network traffic data and find anomalies outside of any of the clusters using a k-means algorithm Dragos, 2022. On the other hand, clustering methods may find it difficult to manage high-dimensional data and may need extensive preprocessing to generate consistent outcomes.

One kind of neural network sometimes referred to as an autoencoder could learn data compression and rebuilding on its own. Their power to spot data objects that greatly depart from the taught patterns adds enormous value for anomaly identification. By means of sensor data and anomaly detection from normal operation, autoencoders can be employed in SCADA systems to identify FDI assaults (Goh et al., 2017). Analysis of sensor data facilitates one to achieve this. Real-time detection can be difficult to attain as autoencoders demand a lot of data for training and can be computationally expensive.

SVMs: one-class models Especially helpful for anomaly detection, a one-class support vector machine (SVM) is a modification on the traditional SVM method. Those working on machine learning developed this method. Most of the data points are surrounded by a border, hence any data point discovered outside this barrier is considered as an anomaly. They create the limitations with this approach. SCADA systems employ one-class support vector machines (SVM) to spot odd trends in sensor or network traffic (SANS, 2023). SVMs will assist with this. On the other

hand, one-class support vector machines (SVMs) find difficulty with high-dimensional data and may hence need significant modification to yield reliable results.

Reinforce Learning Models

Often referred to as RL, reinforcement learning is a kind of machine learning whereby an agent is trained to make decisions using incentives for desired behavior and penalties for non-desired behavior. Another appellation for this type of instruction is RL. Positive and negative reinforcement are combined in this method of teaching. SCADA systems can use reinforcement learning to build adaptive intrusion detection systems capable of learning from their environment and progressively increasing their efficacy. One can create these systems using SCADA tools. One way to define these systems may be adaptive intrusion detection systems. For instance, an RL-based intrusion detection system (IDS) might be taught to spot and react to cyberattacks using incentives for accurate classifications and penalties for both false positives and false negatives (Yin et al., 2020). This would ensure the IDS could notice and respond appropriately for intrusions. Thus, one might learn the intrusion detection system (IDS) to spot and manage cyberattacks. Conversely, virtual reality needs a lot of computer resources and could be challenging to accommodate into real-time control and data acquisition (SCADA) systems. In terms of technology, virtual reality is really new.

Evaluation Tools for Models

Standards of the Model About Evaluation

Usually, performance of machine learning models designed for SCADA intrusion detection is defined by accuracy, precision, recall, and F1 score. Usually applied to assess the performance of the models are these criteria. Accuracy is the correctness of classifications; precision is the ratio

of actual positives among all positive classifications. Precision and accuracy differ in several ways. Measuring using the proportion of accurate classifications can help one to find accuracy. The harmonic mean of the values of precision and recall is F1. Recall is the proportion of real positives found in respect to the total count. The F1 score is harmonic mean of accuracy and recall scores. One such example of this would be the 2020 findings of Yin et al. on research. In this work, an F1 score of 0.97 and an accuracy of 98% random forest model found FDI attacks in a simulated SCADA system.

These steps could not provide sufficient data regarding the performance of machine learning models for SCADA systems. This is true because false negatives—that is, the incapacity to detect an attack—can have rather severe and damaging consequences. Researchers have proposed new measurements to help to tackle this issue. These steps address the false alarm rate—that is, the frequency of false positives—and the detection latency, hence determining the time the model takes to see an assault (Khraisat et al., 2020). These steps are supposed to help to solve problems related to issues. Every one of these tests includes elements pertaining to the false alarm rate and the detection delay as well. In a 2016 publication, Adepu and Mathur used an autoencoder model with a false alarm rate of less than one percent and a detection delay of less than one second to detect foreign direct investment risks in a water treatment plant. This is one example of how an autoencoder model may spot attacks directed against foreign direct investment.

Issues of Model Selection

To choose the machine learning model most suited for SCADA intrusion detection, one has to give numerous elements great attention. Accuracy, computational complexity, interpretability, and real-time performance are among the components in this category; nonetheless, they are not the only possibilities available to utilize. Although deep learning models like autoencoders

and recurrent neural networks (RNNs) have great potential for accuracy, it is quite conceivable that they are too computationally costly for real-time detection in SCADA systems with limited resources. This is the issue even if these models can reach quite high degrees of accuracy.

Researchers have suggested hybrid solutions combining the features of several machine learning models as means of overcoming these obstacles. One hybrid method may be one in which the first classification uses a neural network and the identification of fine-grained anomalies (IBM X-Force, 2023) makes use of a decision tree. Another approach is using ensemble techniques—which combine several models to boost robustness and accuracy—Yin et al., 2020—also provide. Two kind of ensemble techniques are random forests and gradient boosting.

Application in the Real World Practically

Many SCADA systems have found effective application of machine learning models to support intrusion detection. Using autoencoders with a 99% accuracy rate and a detection time of less than one second, Goh et al. 2017 identified foreign direct investment (FDI) threats. The study was conducted within a simulation of a water treatment plant. The F1 score in a 2019 Kim and Kang paper employing RNNs to identify anomalies in data relevant to power grids was 0.95.

Apart from their use in academic research, operational SCADA systems find use in machine learning models. The United States Department of Energy (DOE) has developed an intrusion detection system (IDS) based on machine learning for its power systems. This IDS integrates support vectors machines (SVMs) with decision trees to detect intrusions in real time (DOE, 2021). Using autoencoders, the European Network of Transmission System Operators for Electricity (ENTSO-E) has also developed an artificial intelligence-based intrusion detection system (IDS) in search of anomalies in the data of the power grid (ENTSO-E, 2022).

Directions forward

At this point, the issue of machine learning-based intrusion detection in SCADA systems is still under development and offers several quite intriguing prospects for next research. Investigating explainable artificial intelligence (XAI) methods is one of the subjects of present work. These techniques increase the visibility of machine learning models by helping one to grasp their decision-making mechanisms, therefore enabling simpler interpretation of the generated results. Conversely, as federated learning allows machine learning models to be trained on dispersed data sources without violating the data confidentially, it offers a third area of interest. This stimulates additional interest as well. Finally, scientists are thinking about developing adaptive intrusion detection systems (IDS) competent of learning from their surroundings and improving their efficacy with time by means of reinforcement learning. This is meant to increase resistance against invaders. This kind of activity allows IDS to develop.

4.3 Adversarial Machine Learning in SCADA Systems

Originally a quite young academic discipline, "Adversarial Machine Learning," has lately evolved to probe the sensitivity of machine learning (ML) models to hostile attacks. In SCADA systems specifically, AML installation is crucial since attackers can utilize these weaknesses to conceal themselves, change the system's behavior, or possibly inflict physical damage. Driven by machines, intrusion detection systems (IDS) become ever more crucial in protecting SCADA systems from intrusions. This is true as one can see and stop cyberattacks thanks to intrusion detection systems. Still, bad actors can use these systems by deliberately changing inputs to let machine learning algorithms carry out actions they shouldn't be able to. Adversarial machine learning will be discussed in what follows together with various countermeasures and how it influences SCADA security.

Understanding Opponents' Machine Learning

In adversarial machine learning, bad actors are generating deliberately modified data for ML models. Adversarial examples are inputs aimed to trick the model into generating false classifications or predictions. Imagine a SCADE system here. While an assailant might quietly change the sensor data to pass for an ML-based IDS, in fact the data has been changed to inflict damage. Critical infrastructure problems are especially prone to adversarial instances since human observers could find it difficult—if not impossible—to differentiate from actual facts.

Originally developing the idea of adversarial machine learning using image identification, researchers demonstrated that even little changes to a picture could force a deep learning network to wrongly identify it (Goodfellow et al., 2015). AML methods have then been used in many other fields including natural language processing, speech recognition, and cybersecurity. Since ML models are utilized in SCADA systems to identify minor irregularities suggestive of a cyberattack, AML is a main concern there. Should an assailant fool these models, they could travel almost invisible toward their goals.

Several Types of Adversarial Attacks

Poisoning attacks and evasion attacks are two primary types that would enable us to organize adversarial attacks on ML models.

Attacks in the inferative phase are those of evasion. The attacker generates inputs in this phase meant to evade the machine learning model's detection. This would be an attack changing sensor data in a SCADA system to make it seem normal to a machine learning-based intrusion detection system (IDS), while the data has been changed to inflict damage. Particularly detrimental in SCADA systems are evasion attacks—which can be employed for covert deception—since they

help one avoid detection. This clarifies their great mortality. For example, Mo et al. (2014) painstakingly created the injected data to imply consistent activity, therefore enabling false data injection (FDI) attacks to hide from traditional intrusion detection systems (IDS).

Attacks of Poisoning: Poisoning attacks arise throughout the training phase aimed to corrupt the model. Among these attacks is the one whereby the assailant introduces destructive data into the training ground. For instance, an assailant might inject erroneous sensor readings into the training data needed to teach a machine learning-based intrusion detection system (IDS), therefore generating the wrong patterns in the model. These poisoning assaults especially cause a risk in SCADA systems since they could compromise the integrity of the machine learning model, therefore lowering the model's capacity to detect actual hazards. Biggio et al. (2012) showed that even in circumstances when just a tiny portion of the training data was compromised, poisoning attacks might drastically degrade the performance of machine learning models. This was true even in circumstances when the models' performance remained unaltered.

Methods for Opposing Attacks

One can produce hostile models and launch aggressive attacks in numerous methods. These techniques fall mostly under either white-box assaults, in which the attacker knows everything there is to know about the machine learning model, or black-box attacks—where the attacker knows very little, if anything.

White-box assaults are ones whereby an aggressor has entire access to a machine learning model including all of its parameters, architecture, and training data. Methods such as gradient-based optimization let an assailant produce quite potent adversarial samples. Using the gradient of the loss function, the Fast Gradient Sign Method (FGSM) creates adversarial samples among one of

the most well-known white-box attack strategies (Goodfellow et al., 2015). Here we witness a white-box attack plan in operation. Since white-box attacks allow the assailant to target the defects in the machine learning model, they can particularly be harmful in SCADA systems.

Black-box attacks replace private knowledge of the machine learning model by depending on observable inputs and outputs to produce adversarial examples using the information the adversary has access to. Within the paradigm of black-box assaults, approaches featuring transferability—that is, utilizing adversarial instances generated for one model to attack another—are relatively prevalent. Before assaulting the target model using a surrogate model, a malevolent actor might provide adversarial examples, for instance (Papernot et al., 2017). This would be done prior to the launch of the attack. Black-box assaults, which are more challenging to implement in SCADA systems, can nevertheless be carried out provided one already knows enough about the behavior of the system.

consequences for SCADE Safety

Adversarial machine learning particularly in relation to machine learning-based intrusion detection systems (IDS) substantially affects SCADA security. Should an opponent be able to fool a machine learning model, they will be free to follow their objectives free from discovery and dodge detection without respect for learning. This could have major effects on important infrastructure since ignorance of a cyberattack could result in physical damage, disruption of activities, or even death.

Lack of resilience many machine learning models have is one of the most important problems in security against adversarial attacks. Machine learning algorithms could not behave as predicted given hostile samples even if they were trained on pristine and orderly data. With

SCADA systems, this is somewhat difficult since the data is usually noisy and scattered and a false negative—that is, failing to detect an attack—may have disastrous results. Sometimes these systems also produce insufficient data.

Still another issue calling for attention is the challenge in spotting hostile cases. Standard techniques are hard to spot since hostile samples are created to seem exactly like real data. While the data has been changed to cause damage, an attacker would be using sensor data in a SCADA system to make it seem normal to a machine learning-based intrusion detection system (IDS). To identify these little changes, one must use advanced techniques able to find trends suggestive of hostile activities.

Reducing Agent Strategies:

Several approaches have been suggested to reduce the adversarial machine learning related hazards in SCADA systems. Three general categories will allow one to arrange these methods: strong model creation, adversarial training, and anomaly detection.

robust model construction: By producing naturally resistant to hostile examples using machine learning models helps one to minimize the consequences of adversarial attacks. One method one may use to do this is regularizing, which penalizes complicated models and reduces their probability of overfit conditions. For example, a 2018 work by Madry et al. shows that adversarial training—in which the model is taught on a mix of clean and adversarial data—may significantly increase its resilience to adversarial attacks.

Teaching the machine learning model on hostile situations through adversarial training helps it to identify and respond to such events. This method has been demonstrated to raise the resistance of machine learning models against adversarial attacks, hence facilitating their enhanced self-

defense. Results of a 2015 Goodfellow et al. study show that adversarial training—for example—appears to dramatically diminish the rate of successful evasion attempts on deep learning models.

One can further reduce adversarial attacks by use of anomaly detection technology to identify antagonistic examples. A fundamental part of anomaly detection is analyzing the inputs to the machine learning model and finding trends distinct from the expected behavior. Regarding photo recognition tasks, for example, a 2017 Grosse and colleagues study shown how anomaly detection methods might identify dangerous environments. Little differences in sensor data observed in SCADA systems could indicate an enemy attack.

Useful Cases in Practical Applications

Adversarial machine learning becomes a rising problem in real-world SCADA systems since a successful attack on them might have disastrous consequences. One instance of this is the 2014 work by Mo et al., which meticulously generated the injected data to mirror normal behavior, therefore allowing conventional intrusion detection systems (IDS) to avoid false data injection (FDI) assaults. Studies by Biggio et al. (2012) indicate that poisoning attacks could seriously compromise the performance of machine learning models. This is the actual regardless of the small percentage of the training data taken under account. This point of view was validated by data provided by the situational analysis.

Working diligently, experts in the industry are developing new solutions to guard SCADA systems against recognized hazards. The Department of Energy (DOE) of the United States of America developed an intrusion detection system anchored on machine learning to protect power systems from potential threats. Adversarial training, according to the Department of Energy (2021), is a method an intrusion detection system uses to increase its defensive capability against

hostile activity. Analogously, the European Network of Transmission System Operators for Electricity (ENTSO-E) has created an Intrusion Detection System (IDS) designed to identify unexpected occurrences in grid-generated data. This method was developed with a view toward preserving data authenticity. According to ENTSO-E (2022), machine learning and anomaly detection are the means by which this purpose is accomplished.

Future Directions of Methodologies

Although adversarial machine learning is still developing, future scientists could find numerous intriguing routes to follow. One field of research is the evolution of explainable artificial intelligence (XAI) methods. These methods are more clear and interpretable of their findings since they enable one to understand the decision-making procedures of machine learning models. Still another area of interest is the use of federated learning, which lets machine learning models be trained on distributed data sources without endangering data security. Lastly, researchers are looking at ways to design adaptive intrusion detection systems (IDS) capable of learning from their environment and, using reinforcement learning, increasing their efficacy over time.

4.4 Real-Time Detection Challenges in SCADA Systems

Real-time intrusion detection is a required need for SCADA systems to operate as they should. Cybersecurity is therefore very crucial since delays in identifying and reacting to cyberattacks can lead to serious consequences including physical damage, operational difficulties, and even death. This is so because, although the most often occurring kind of cyberattack, cyberattacks can entail physical attacks in addition to digital ones. Machine learning (ML), which has shown to have great promise, can significantly raise the capacity of intrusion detection systems (IDS) in SCADA environments. Currently, this has been discovered. On the other hand, using real-time SCADA

systems with intrusion detection systems grounded on machine learning poses several significant challenges. One can sort these difficulties into several groups. Just two of the requirements that have to be satisfied are low false alarm rates and great precision. Other restrictions include data latency, computational limitations, and high accuracy demand. This part will include a thorough examination of these difficulties together with a discussion of the several possible remedies that might be used to allow real-time intrusion detection in SCADA systems. The aim of this part is to present both of these aspects.

computational limitations

The limited availability of computational resources in industrial environments is one of the most important obstacles in the way real-time machine learning-based intrusion detection systems may be included into SCADA systems. Usually limited in processing capacity, memory, and storage capacity, SCADA systems most of the time depend on outmoded hardware and embedded devices. This is true considering SCADA systems are continually changing. These limitations make using machine learning models requiring a lot of computational resources difficult—such as deep neural networks (DNNs), which demand a lot of resources for training and inference. Getting these models operating also needs significant expertise.

One instance of this is the 2019 study Kim and Kang conducted showing, with great accuracy, recurrent neural networks (RNNs) discovered anomalies in data relevant to power grids. Real-time implementation of recurrent neural networks (RNNs) proved insufficient for devices with low resources due of processing load. Researchers have developed lightweight machine learning techniques and edge computing systems able to analyse data locally and offer real-time detection capabilities in order to overcome this difficulty. These remedies are designed to fit the intricacy of the problem. Studies by Khraisat and associates in 2020 revealed that instead of deep neural

networks (DNNs), SCADA systems often employ decision trees and support vector machines (SVMs). This is thus because faster inference times and less processing demand these two types of models.

Model compression methods provide even another way to address the issues resulting from compute limits. Among these techniques are knowledge distillation, quantitative scaling, and trimming. These methods enable one to reduce the size and complexity of machine learning models; this is accomplished without appreciable model performance loss. A 2016 Han et al. study claims, for instance, pruning and quantising can help to reduce a deep neural network (DNN) by as much as 90% even while preserving accuracy. Teaching a smaller "student" model to mimic the conduct of a bigger "teacher" is known as knowledge distillation. This method corresponds with another in a similar line. Hinton et al. (2015) claim this lets one use in real time on devices with specific limited resources.

timeliness and data lagencies

Real-time intrusion detection for SCADA systems calls for fast data processing from sensors, controllers, and other devices. If the system is to accurately detect intrusions, it must meet this criterion. Conversely, frequently producing vast volumes of high-frequency data by statistical control and data acquisition (SCADA) systems can cause processing delays and extra data latency. For voltage, current, and frequency, a power grid could be able to generate for many thousand thousand samples per second data. Using the traditional machine learning techniques in real time analyses of these data becomes difficult this way.

Experts have suggested implementing Apache Kafka and Apache Flink using streaming data processing platforms in order to address issue. The Apache Foundation (2023) claims these

systems can control fast data flows and offer real-time insights. These techniques enable machine learning models to manage created data as it comes from sources. This influences intrusion detection timeliness and helps to lower latency. One advantage of it is that one may manage data as it is produced. A 2020 Yin et al. random forest model deployed on Apache Kafka found in a simulated SCADA system the detection delay was less than one second. This was shown by the model's capacity to yield this outcome.

This is yet another method that might significantly lower the latency times of data. One way that data might be handled locally on edge devices instead of being routed to a central server for analysis is edge computing. Computing on the edge of a network greatly lowers data movement times. This also allows real-time identification even in situations of a limited network bandwidth. Data from Goh et al. (2017) offers evidence of this. This investigation shown that at a water treatment facility an autoencoder model housed on edge devices may get a detection latency of less than 500 milliseconds.

False Alarms: Precision vs Rates

Without low rate of false alarms and great degree of precision, SCADA control systems cannot detect breaches in real time. This helps to lower the disturbance caused by false positives and raises the likelihood that real threats will be discovered. Conversely, the noisy and fragmentary character of SCADA data as well as the sophisticated and always shifting tactics attackers use make great accuracy and low false alert rates difficult. This renders both of these targets difficult.

One instance of this would be covert deceptive attacks, sometimes referred to as SDAs. These attacks are meant to represent legal behaviour, so conventional machine learning methods are particularly challenging in their search. False data injection (FDI) assaults could exactly match

the injected data to resemble normal activity, therefore shielding themselves from detection by conventional intrusion detection systems (IDS), according an analysis by Mo et al. (2014). Researchers have suggested the application of creative machine learning techniques to address this problem. An adversarial training and anomaly detection are two such approaches. These systems can identify even minute trends implying hostile activity.

An anomaly identification in machine learning is the process of looking over the inputs to the model and spotting anomalies from the usual behaviour. The basic process of the operation is this. Goh et al. (2017) for example examined how well autoencoders could identify anomalies in sensor data, therefore highlighting targets for foreign direct investment (FDI) in a water treatment plant. The sensors generated automated data analysis that assisted to reach this. Adversarial training is a related strategy whereby the machine learning model is taught on hostile environments to improve its resilience to defensive strategies. This helps to boost the capacity of the model by raising its resistance against attacks. This is why go over Goodfellow et al. (2015) findings. Adversarial training notably lowers the successful evasion attempts on the rate of deep learning models.

Using ensemble techniques is another way one could try to increase accuracy and lower the false alert count. These methods mix many machine learning models to improve performance. For the identification of foreign direct investment (FDI) dangers in a simulated SCADA system, Yin et al. (2020) investigated and found, for instance, a random forest model obtained an accuracy of 98% and a false alarm rate of 0.1%. As it occurred, this was accurate exactly. Analysing anomalies in power grid data, Kim and Kang (2019) found that an ensemble of recurrent neural networks (RNNs) and support vector machines (SVMs) rated an F1 of 0.95. Combining the two forms of neural networks allowed this.

Scalability and Change

Usually, SCADA systems are rather large and advanced. Spread over great lengths of land, they consist of hundreds of sensors, controllers, and other equipment. Such systems should feature scalable machine learning models that are also flexible enough to fit evolving conditions so as to identify intrusions in real time. Conversely, the dynamic and chaotic character of SCADA systems makes obtaining scalability and flexibility difficult.

The condition of the surroundings, the degree of equipment wear and tear, and the operational changes could affect the behaviour of a SCADA system over time. These advancements risk influencing the performance of machine learning models, thereby influencing accuracy and maybe increasing the false alarm count. Online learning is one adaptive machine learning approach proposed by academics to meet this demand. Still still another adaptive machine learning method is transfer learning. These techniques enable models to evolve with the times and learn from just acquired data.

Online learning enables the machine learning model to be updated in real time as new data becomes available, therefore enabling its capacity to change with the system. Regarding identifying abnormalities in streaming data from SCADA systems, for instance, Gama et al. (2014) found that online-based learning systems could have quite high degree of accuracy. The studies revealed this. Conversely, transfer learning is the method of sharing knowledge from one field into another. This helps the machine learning model to apply what it has learnt in fresh surroundings. For instance, based on 2010 Pan and Yang study on anomalies in data connected to electrical grids, transfer learning has the potential to enhance the performance of machine learning models.

Furthermore, enabling scalability and adaptability is the application of distributed computer architectures. Among such models are Apache Hadoop and Apache Spark. By enabling independent examination of enormous volumes of data in parallel, these models support machine learning models to be able. One example of this is the 2016 research by Zaharia et al. showing Apache Spark could distribute the processing load over several nodes, hence enabling real-time intrusion detection in large-scale SCADA systems.

Collaboration with Current Systems

Still another key problem is the system's incapacity to combine current SCADA systems with intrusion detection systems with machine learning orientation. Including modern machine learning techniques into SCADA systems could reveal to be a difficult dance. This can be troublesome as most SCADA systems rely on proprietary protocols and obsolete hardware. The SCADA system has to remain operational as a last stage of the integration process and the machine learning-based intrusion detection system has to not create any more vulnerabilities. This has to be met. This need has to be met right now..

Researchers have developed middleware solutions acting as a link between the SCADA system and the machine learning-based intrusion detection system (IDS) so enabling connection between the two systems. Using a middleware layer could help to merge older SCADA systems with machine learning-based intrusion detection systems (IDS) more easily (Cardenas and colleagues, 2011). Results of the investigation validated this. theoretically, this integration should be feasible without fundamentally changing the current configuration. Open standards like OPC UA (Open Platform Communications Unified Architecture) will help to enable the integration of SCADA systems with intrusion detection systems depending on machine learning (OPC

Foundation, 2023). One must offer a universally applicable communication interface if one is to reach this aim. This point reminds me in several respects the one that preceded it.

Applications from the Real World

Real-world SCADA systems have included machine learning-based intrusion detection systems to instantly spot invasions. Notwithstanding the challenges thrown forth, this has been achieved. Developed for power grids, the United States of America's Department of Energy (DOE) created a machine learning intrusion detection system (IDS) To improve its detection powers, this system uses decision trees and support vector machines (SVMs). This technology was intended to detect automatically occurring invasions. Based on machine learning and autoencoders to detect anomalies in the data gathered from the power grid, ENTSO-E, or the European Network of Transmission System Operators for Electricity, has created an intrusion detection system (IDS) (ENTSO-E, 2022).

Along with these major installations, smaller SCADA systems—such as those found in water treatment plants and oil refineries—have made use of machine learning-based intrusion detection systems. Conversely, Adepu and Mathur (2016) found that in a water treatment plant with a detection delay of less than one second, an intrusion detection system grounded on machine learning was able to identify foreign direct investment concerns. With autoencoder models, one could claim the same; Goh et al. (2017) showed they could detect 99% of anomalies in the sensor data taken from oil refineries.

Future Directions

Though real-time intrusion detection in SCADA systems is still under development, there are several interesting avenues that should be looked at to enable research activities to be conducted.

The creation of tools for explainable artificial intelligence (XAI) is one of the fields of study under increasing focus nowadays. These approaches help to clarify the decision-making processes of machine learning models, therefore raising the interpretability of the output for such models and increasing the visibility of the models. Conversely, the implementation of federated learning—which enables machine learning models to be trained on scattered data sources without endangering the privacy of the data—draws increased interest. Long lastingly, scientists are looking at using reinforcement learning to create adaptive intrusion detection systems (IDs) able to learn from their environment and increase their efficiency over time. We take this measure to build IDS.

4.5 Time Series Analysis for Intrusion Detection in SCADA Systems

Since they produce massive amounts of time-stamped data from sensors, controllers, and other devices, time series analysis is a basic element of intrusion detection in SCADA systems. This is so as SCADA systems generate a lot of data. Natural chronological character of the data in issue includes observations of temperature, pressure, voltage, and flow rate. Therefore, the timing of the data points as well as their sequence define two of the most important factors to consider while trying to grasp the behavior of the system. Machine learning models based on time series analysis help to detect cyber threats including stealthy deception attacks (SDAs), denial-of- service (DoS), and fake data injection (FDI). We refer to these forms of cyberattacks as "attacks" collectively. These techniques allow one to identify patterns and anomalies in the provided data. Within the framework of this part, we explore the function of time series analysis in SCADA intrusion detection, the issues connected with the analysis of time series data, and the machine learning approaches applied to solve these problems.

Value of Time Series Analysis for Systems of Control

Made to generate continuous streams of time series data, statistical control and data acquisition (SCADA) systems allow one to monitor and control industrial operations in real time. This guarantees perfect monitoring and control of operation. This data provides essential knowledge about the system's running state, which helps operators to spot abnormalities and respond correctly to any potential risk. When compared to a consistent divergence from predicted data, say a fast rise in temperature or pressure could point to a malfunction or a cyberattack. Conversely, consistent deviations from predicted values can indicate a concealed attempt at deceit.

Time series analysis is highly important for this purpose since FDI assaults—which involve the manipulation of sensor data to deceive operators and disrupt operations—also depend on this. FDI assaults are quite severe since they include sensor data manipulation. An opponent might modify the temperature readings of a power plant, therefore overloading the equipment of that facility. They might so similarly alter the pressure readings in a water treatment plant to induce leaks. Both of these pictures depict how an enemy may cause trouble. Usually fitting the expected range of normal activity, traditional intrusion detection systems (IDs) depending on stationary rules or signatures may find it difficult to identify these little changes. This is extremely true as most individuals agree that these deviations lie within the usual spectrum of activity. On the other hand, machine learning techniques include time series analysis can find trends and patterns in the data suggestive of detrimental behavior even in circumstances when the individual data points seem to be good. This is so since these methods let the data to be different from one another.

Challenges with Time Series Analysis

Time series data analysis in SCADA systems has certain difficulties that must be overcome. These difficulties show in too high dimensionality, noise, and missing data. These difficulties

make it more difficult to find anomalies and raise the possibility of false positives and false negatives emerging.

Among other things, sensor readings can be greatly influenced by mechanical vibrations, ambient circumstances, and electromagnetic interference by a large range of elements. Moreover, systems of control and data collecting (SCADA) are commonly applied in situations defined by too high noise level. Since this noise could hide the underlying trends in the data, machine learning systems could find it difficult to distinguish between normal changes and anomalies. For the algorithms in pattern recognition, this would make extra challenge. For instance, noise in sensor data could reduce the accuracy of autoencoder models, according a 2017 Goh et al. study. This was seen from the standpoint of assessing the risks related to foreign direct investment (FDI).

Missing Data The lack of data in SCADA systems is another regularly happening and maybe problematic situation. This could occur should links of communication break or if sensors malfunction. Lack of data could cause time series to show gaps, therefore complicating the trend analysis and odd event identification. This is true since information is lacking shown by gaps in the time series. For a water treatment plant lacking data, Adepu and Mathur (2016) found, for example, that machine learning-based intrusion detection systems (IDS) become less effective in spotting attacks linked to foreign direct investment (FDI).

SCADA systems generate in many situations high dimensionality time series databases. Usually, these systems are made of hundreds or thousands of highly frequency reading capable sensors. This tremendous complexity could make it challenging for models of machine learning to effectively understand the data and spot significant trends. For example, Kim and Kang (2019) discovered in a high-dimensional power grid data set analysis challenges for recurrent neural networks (RNNs) in terms of anomaly detection. This proved to be applicable.

Methods of Machine Learning for Time Series Analysis

Many ML techniques have been developed to overcome challenges in SCADA systems' time series analysis. These techniques can be mostly categorized as standard statistical techniques, classical ML algorithms, and deep learning models.

To perform time series analysis, SCADA systems widely use statistical techniques that have been available for a considerable length of time. Among the several techniques are autoregressive integrated moving average (ARIMA) and exponential smoothing. These techniques allow one to find anomalies by correlating observed values with expected values, therefore defining the temporal relations in the data. This analogy enables one to spot irregularities. This enables one to ascertain whether the standard is followed exactly or otherwise. As shown by a 2018 Hyndman and Athanasopoulos (2018) analysis, ARIMA models were ultimately able to identify anomalies in the data by spotting deviations from predicted patterns in power grid data. This work demonstrated ARIMA model-based identification of certain aberrations.

Classic ML Algorithms: Conventional machine learning techniques can also be used advantageously in time series analysis in SCADA systems. Machine learning has still another application here. Among the various approaches in this area are k-nearest neighbors (k-NN), support vector machines (SVMs), and decision trees. Extraction of features from time series data is one widely utilized application of these routinely applied methods in practice. These features combine temporal patterns (such as trends and seasonality) with statistical measurements (such as the mean and the variance). For example, Yin et al. (2020) showed remarkable degree of accuracy there after looking at how effectively a random forest model found possible dangers to foreign direct investment. Statistical analysis of acquired sensor data helped to do this by means of extracted features.

Deep learning models have been quite well-known over the past few years because to their ability to spot minute trends in time series data. Among the various models that meet this category are convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and recurrent neural networks (RNNs). Especially helpful for SCADA systems, these models allow to add temporal correlations and manage high-dimensional data. These qualities make them particularly worthful. One such are studies Kim and Kang (2019) showing high degree of accuracy LSTM networks found anomalies in data pertaining to power grids. This was attained by means of long-term dependency modeling in the time series.

Applying time series analysis in SCADA intrusion detection

A wide range of cyber hazards in SCADA systems including FDI attacks, DoS attacks, and SDAs has been successfully identified using time series analysis. These applications highlight how effectively ML models may enhance the security of important infrastructure by identifying minute trends and anomalies in time series data.

False Data Injection (FDI) Attacks: Using FDI attacks—which involve sensor data manipulation—one can interfere with operations and deceive operators into believing something false. Time series analysis can assist to identify these attacks by pointing up variations from the expected trends in the data. One can reach this using different pattern recognition. Time series analysis, according to Mo et al. (2014), can identify foreign direct investment (FDI) hazards by pointing up voltage and frequency data anomalies. The framework of power grid data helped to show this. Time series analysis turned out to be the vehicle used in this regard. Relatedly, Adepu and Mathur (2016) looked at sensor data trends to demonstrate how autoencoder models

could identify foreign direct investment (FDI) attacks in a water treatment plant. One can do this by means of data analysis derived from the sensors.

Sometimes referred to as denial-of- service (DoS) attacks, DoS assaults disrupt links or flood networks with traffic, therefore depriving operators of monitoring and control of the system. Time series analysis lets one identify these assaults by noticing unusual trends in network traffic or communication delays. By looking at packet arrival times and network delay, time series analysis—for example—can find DoS assaults in SCADA systems, according a 2011 Cardenas et al. paper.

Since Stealthy Deception Attacks (SDAs) are supposed to hide themselves by mimicking real activity, they are somewhat difficult to detect with the conventional approach. Time series analysis enables the identification of these attacks by helping to find minute fluctuations from normal activity over time. Goh et al. (2017) for instance investigated patterns in sensor data using autoencoder models to demonstrate that these models might detect SDAs in a water treatment plant. In accordance with this, Kim and Kang (2019) looked at whether LSTM networks able to foresee long-term dependencies in the time series might discover SDAs in data connected to the power system.

Challenges and Future Directions

Though time series analysis shows promise in SCADA intrusion detection, some problems still require addressing. One of the most important concerns is the need of obtaining high-quality training data since machine learning models depend on large volumes of labeled data to comprehend the trends of normal and detrimental behavior. Given the sensitive quality of the

data and the limited access to real-world attack data, obtaining tagged data in SCADA systems can be difficult. This may complicate the operation.

Real-time detection is another challenge to handle since delays in seeing and resolving intrusions could have major consequences immediately. Using machine learning models depending on advanced algorithms, such deep learning, real-time detection using these models may prove challenging since these models may require a great volume of computer resources and time to process data. To solve this, researchers have presented lightweight machine learning algorithms and edge computing systems suitable of processing data locally and offering real-time detection capacities.

Another interesting road for future research is here as adversarial training and anomaly detection mix time series analysis with other machine learning techniques. For example, time series analysis combined with anomaly detection methods could improve the capacity of machine learning models to spot little behavioral deviations from the norm. Similarly, time series analysis combined with adversarial training offers a chance to increase machine learning model resilience against attacks.

Application in the Real World Practically

Time series analysis has been effectively used to provide real-world SCADA systems improved intrusion detection capability. Designed for use with power grids, an artificial intelligence-based intrusion detection system (IDS) using time series analysis to identify anomalies in voltage and frequency data has been created by the United States Department of Energy (DOE, 2021). Using time series analysis to find anomalies in data pertaining to the power grid, the European Network

of Transmission System Operators for Electricity (ENTSO-E) has also developed an artificial intelligence-based intrusion detection system (IDS).

Apart from these major worldwide installations, smaller SCADA systems—such those used in water treatment plants and oil refineries—have also made use of time series analysis. Claims a 2016 Adepu and Mathur study, time series analysis can find foreign direct investment (FDI) attacks in a water treatment plant with a delay of less than one second. In a related line, a 2017 Goh and colleagues study shown that autoencoder models could identify anomalies in 99% correct sensor data generated from an oil refinery.

Conclusion

Time series analysis is a required component of SCADA systems since it helps machine learning models to detect trends and abnormalities in time-stamped data. Strong detection calls for this essential element. Companies can raise their ability to identify cyber hazards such FDI assaults, DoS attacks, and SDAs by means of traditional statistical methodologies, conventional machine learning algorithms, and deep learning models. Issues such noise, missing data, and too high dimensionality must be addressed in the meantime to ensure that time series analysis within real-world SCADA systems is successful. Research should largely focus on advancements in the availability of high-quality training data, the development of lightweight machine learning algorithms, and the integration of time series analysis with other machine learning methodologies.

Chapter 5: Data Acquisition and Modeling Methodology

5.1 Introduction

In today's technology-driven world, maintaining the safety and performance of SCADA networks has moved to the top of virtually every organization's list of priorities. SCADA systems are essential for running and managing important infrastructure facilities such as power plants, water treatment centers and transport systems. Because they are so important, these systems are now frequently attacked by cybercriminals. By applying machine learning solutions, the approach in this research aims to improve how intrusions are identified in SCADA networks.

5.1.1 Context and Motivation

The idea for the SCADA system began in the 1960s, when computers became advanced enough to permit real-time controlling of industrial operations. Ever since they connected to the internet, these systems helped achieve better remote surveillance, predict how they might break with current maintenance and work more smoothly. Even so, linking SCADA networks to other systems made them more at risk from cyber-attacks aimed at critical national infrastructure.

The 2010 Stuxnet attack showed how SCADA vulnerabilities can cause damage to physical systems, for example by causing harm to an Iranian nuclear facility. Because SCADA environments are often made up of both old and new technologies, security specialists face distinct difficulties. Most traditional security products are not designed to meet the particular requirements of SCADA networks.

As data being collected expands and cyber threats become more active, people are realizing why rule-based IDS solutions may not work well in SCADA. They might be very good at blocking threats lawyers have seen, but they usually lack flexibility for alerting lawyers to new attacks. For this reason, machine learning lets organizations benefit from evolving insights drawn from data to enhance their intrusion detection.

5.1.2 Objectives of the Data Collection and Modeling Framework

The main goal of this research methodology is to make use of machine learning to design a more supportive and nimble intrusion detection system for SCADA systems. It is designed to accomplish certain important objectives.

- Turning Data into Insights: Thank to SCADA, we convert the large amount of collected data into meaningful info to detect safety hazards.
- Learning: Design a security system capable of updating itself when exposed to different types of threats.
- Lessening of False Alarms: Traditional IDS tend to indicate suspicions about normal activities because their rules are not relaxed enough. A major objective is to reduce false positives so that alerts are important and worthwhile to act on.
- Real-time detection is a main goal of this approach, making it suitable for operations that SCADA handles. If you notice an incident quickly, you can take action right away.
- As one looks at cyber security risks, the approach also considers system health, worker behavior, and environmental factors. This kind of study helps us to more precisely identify the hazards in relation to security issues.

- Simple Integration with Existing Equipment: The aim is to easily mix the new solutions into these existing surroundings since most SCADA systems are created with a combination of old and modern technologies.
- . it should not stop at applying machine learning; always assess their applicability, modify the models, and progressively strengthen the system.

5.2 Technical Overview of the SCADA Environment

Understanding their structure, the way data is transferred, and the possible risks they may run against becomes more and more important as we advance into SCADA systems. One must have a thorough awareness of the features, structure, and possible flaws of these fundamental systems if one is to ensure their durability and strength.

5.2.1 SCADA System Architecture and Functional Components

The architecture depends on both closely connected hardware and software to offer the remote monitoring and management of a wide range of industrial processes. The Master Terminal Unit (MTU) determines a great part of the system's capability. Apart from being the main controller, the MTU also gathers all the data from different field equipment. Then it uses the knowledge it has to convey the results to human operators so they may guide their activities.

MTUs also show up often alongside Programmable Logic Controllers (PLCs) and Remote Terminal Units (RTUs). These kinds of devices are found in many different remote locations and their main use is to translate sensor data into digital form.. The collected data is sent from RTUs and PLCs to the MTU after it is taken. At the same time, HMIs are used by operators to interact with the SCADA system, showing important data clearly and enabling true real-time control.

5.1.1 Communication Protocols and Data Transmission Pathways

Data flowing on a SCADA network is what keeps the entire system working well. At the sensor stage, where environment or equipment status is recorded, the raw data goes through an RTU or PLC and changes from analog to digital. The information obtained from digitized data can be sent via various protocols such as Modbus, DNP3 or IEC 60870-5-104 and these can be fixed or wireless networks.

While the MTU takes in this data, it is processed and kept safe and the key information is always shown on the HMI. At this screen, operators can both read the information and send commands such as changing machine settings or addressing alerts. This bi-directional flow of data and commands ensures that the SCADA system can both monitor and control industrial processes in real-time.

5.1.2 Known Vulnerabilities and Potential Attack Surfaces

While SCADA systems offer unparalleled control and monitoring capabilities, they are not without vulnerabilities. Many SCADA systems, originally designed in an era where security was not a paramount concern, now find themselves connected to broader networks, exposing them to modern cyber threats.

Today's cyber threats are often caused by using old systems. Older systems are more at risk, since they haven't been created with today's cybersecurity rules in place. At the same time, linking older systems to newer ones can lead to unsuspected security issues.

It is possible for attackers to abuse the methods of communication that power SCADA's data flow. Protocols without built-in security, for instance Modbus, leave the systems they serve open to unsecure commands and snooping.

As a result, since SCADA systems are connecting to more networks and the internet, it is easier for threats to reach them. Sloppy security of remote access points can make them easy entry points for cybercriminals.

What we saw with the Stuxnet worm, focused on SCADA systems, reminds us how dangerous it can be when flaws are used by attackers. Because of this attack which stopped an Iranian nuclear project, it became clear what problems could arise because of SCADA weaknesses.

Basically, getting to know the SCADA environment and its safety issues is the main step in protecting it. It supports the creation of advanced security solutions that guarantee these systems remain stable and safe amid modern connections.

5.3 Data Acquisition and Preprocessing Pipeline

In machine learning, you usually find the outcomes you input. The degree of performance of models depends on the quality of the created training data. Particularly SCADA systems make their data essential for monitoring industrial activities and also for identifying any indications of cyber threats. An intrusion detection system depends on precise, dependable, and valuable data collecting to operate as it should.

5.3.1 SCADA Data Sources and Collection Techniques

Leveraging the full possibilities of machine learning in the SCADA system calls for using a range of data sources, each providing special insights on the operation and possible vulnerabilities of the system.

5.3.1.1 Network Traffic Logs

Any system like this depends mostly on the network, thus monitoring the traffic it carries exposes important information on its purposes. Logging their actions as they cross the SCADA network allows one to closely review data packets. Among other information these logs offer are date, source, destination, port number, size, and arrival time. Traffic analysis allows one to identify both regular network behavior patterns and abnormal ones that can suggest cyber-attacks. A denial-of-service attack may start with a leak of a lot of data requests unexpectedly emanating from an unidentified IP..

5.1.1.1 System Logs

Like other advanced digital systems, SCADA systems generate detailed operation logs. System logs show a list of recent events, including starting or stopping the system, errors that occurred, what the users did and any changes to the system's configuration. Such logs enable us to track the well-being and status of the SCADA components. They may also expose access to your system that you didn't grant, changes made to system settings or acts that might point to harmful events. A number of failed attempts to log in at the same moment usually hints at a brute force attack.

5.1.1.1 Sensor Data

Being a main link between physical processes and SCADA management, sensor data informs us in real time of temperatures, pressures, flow rates and other things. In the world of intrusion detection, unexpected readouts from sensors could be a sign of both equipment breakdowns and cyber-feared problems, particularly when cyber-physical attacks are involved. For instance, sensors might notice an unexpected rise in temperature which might happen after a hacker takes over a control valve.

5.3.2 Data Cleaning, Normalization, and Quality Checks

The success of a machine learning project is greatly affected by using quality and regular data. There are unique difficulties and chances in this field, thanks to the many sensors, logs and live data streams in SCADA systems. As a result, preparing this data for analysis by cleaning and normalizing it becomes essential.

In SCADA, there are a number of complications when working with data. Firstly, getting data from the real world means dealing with missing values. Because of challenges with sensors, flaws in computer networks or minor system failures, data streams can sometimes be missing. Experts resort to filling in these spaces with stats techniques or computer models to deal with this. If a temperature sensor goes wrong now and then, the missed values can be filled in with the readings from neighboring sensors or from past trends.

Also, occasionally, small, unstable movements in the data from these systems are mistaken for real changes, when they are not. If required, these algorithms spot outliers and can fix or eliminate them. If ensuring the accuracy of data matters most, unusual results are checked by hand.

However, the main problem for normalization is that data points are often recorded on various scales. A wide range of parameters are watched by SCADA systems, like pressures, flows, temperatures and the speed of rotation. Each can use a different way to measure values. Bringing data of varying scales into a machine learning model can cause some features to be too strongly emphasized in the model's results. Min-Max scaling or Z-score normalization changes each feature so that all numbers are between [0,1] or have a mean of 0 and a standard deviation of 1. This guarantees every feature helps the model to train properly.

5.3.3 Feature Engineering: Extraction and Selection

Even though SCADA systems produce a large amount of data, much of it is not relevant to every problem being investigated. Machine learning models are fed only the information that matters by following the feature extraction and selection processes.

Style transferring means using these styles to change the raw information into a group of features. Another method is to take out some key characteristics such as the regular size of packets, the number of novel IPs or the connection ratio of incoming and outgoing data. They take the raw data and strip it down to its most essential points for use in machine learning.

The process could be applied to SCADA sensor data too. Instead of reviewing temperature data from the boiler every second, a system could collect the daily average, the highest temperature found in a day or the differences in temperature over a day period.

Although the data is extracted, many of its features may not be really significant. It is at this point where feature selection needs to take place. Modelers may use various methods, including correlation analysis, mutual information and recursive feature elimination, to choose which information to keep and which to remove. For example, when two sensors give similar results, it is possible to remove one of them and ignore those results to cut down on how much data we need to look at.

5.4 ML Model Design and Selection

Among the many aspects of machine learning, choosing the right model can be almost as difficult as it is important. As SCADA systems have interesting requirements, selecting the right protocol is more important than ever. Depending on this choice, different intrusion detection approaches are required and the impact can affect both security and how things are operated.

5.4.1 Model Selection Criteria Based on SCADA Requirements

Choosing a machine learning model is far from a one-size-fits-all decision, especially in the context of SCADA systems. Several factors dictate this choice, each interplaying with the others to define the best-fit model for the system's intrusion detection needs.

1. If the data is linear, a linear model is usually chosen, but if not, a model like Neural Networks will be preferable. In the case of data with distinct decision boundaries, decision trees or Random Forests can be the better option.

2. Volume matters: Deeper models are more effective for studying huge data volumes that have lots of entries. On the flip side, lower dataset sizes tend to work better with basic models like Support Vector Machines or k-Nearest Neighbors to help them not overfit.

3. Sometimes, it’s necessary to know how and why a SCADA system alerted its users when evaluating threats or risks. Unlike Neural Networks, Decision Trees or Logistic Regression give more detailed explanations of their conclusions.

4. The System Must Decide Quickly in Attack Time: Real-time intrusion detection calls for fast decision-making. Naïve Bayes and Logistic Regression which use less computing power, may be chosen over models that are more demanding if responding quickly is crucial.

5. Even though how well a model fits the training data is important, how it works on unseen data is even more essential. Cross-validation is a useful method to assess this situation. Models that are accurate and adaptable to new situations without becoming too specialized are what is wanted.

6. Whenever you are asked to classify data into set categories (like regular activity and anomaly), tools like Random Forests and Gradient Boosted Trees are often used. For continuous outcomes, models such as Linear Regression or regression Neural Networks are preferred for regression tasks.

7. Adaptability: In the dynamic environment of SCADA systems, where data patterns can evolve, models that can adapt to changing data dynamics, like online learning models or models that support incremental learning, are beneficial.

8. Model Robustness: Given the security implications in SCADA systems, models that are robust to adversarial attacks or data manipulation hold an edge. Models that can withstand data noise, missing values, or even adversarial attempts to deceive the intrusion detection system are invaluable.

5.4.2 Technical Overview of Selected Machine Learning Models

Many models have grown to compete for roles in SCADA systems because of their usefulness in machine learning. Every school stands out due to its special principles and mechanisms. We'll look more closely at how these models operate.

5.4.2.1 Decision Trees

Basically, a Decision Tree examines the data by making a series of choices until coming to a final decision. Breaking the model into smaller parts builds the structure of a tree, making the mapping from input to output clear and understandable. Because SCADA systems must handle a wide range of data, from numbers to labels, Decision Trees offer a great adaptability advantage. Databases can take in different kinds of data, without always needing to be normalized. Even so, never having many parts can make their performance weak at times. Such trees often become difficult to manage

when they have to deal with the intricate datasets you regularly encounter in SCADA settings. When a model is too intricate, it may store all of the training information without being able to use it elsewhere.

Even though Decision Trees operate using a set of rules (instead of math), you can represent how one rule splits into two by writing:

$$\begin{cases} c\,1 & if\ x \leq \theta \\ c\,2 & if\ x > \theta \end{cases}$$

Where f(x) represents the decision output based on the feature x, θ is the decision boundary (or threshold), and c1 and c2 are the class labels.

5.4.2.2 Neural Networks

If you look at Neural Networks, you'll see their deep variants are built of a lot of layers linked by nodes or "neurons." Thanks to their processing through layers, Neural Networks identify patterns well, even if those patterns may be hidden from human understanding. Finding out how things are connected is crucial in SCADA, given that tiny clues may indicate an upcoming system failure or information leak. Yet, possessing all these skills isn't free. The strong performance promised by Neural Networks comes at the cost of keeping their workings invisible. Since in SCADA even one wrong decision can cause serious consequences, the lack of openness can be both useful and harmful. Besides, the fact that they demand a lot of computational power while training could cause issues for applications that must respond quickly.

The output for a basic neural network with a single neuron can be expressed as:

$$y = \sigma(w \cdot x + b)$$

Where:

- y is the neuron's output.
- σ represents the activation function (e.g., sigmoid, tanh, ReLU).
- w is the weight vector.
- x is the input vector.
- b is the bias.

5.4.2.3 Support Vector Machines

Among different classifications, Support Vector Machines (SVM) attract attention with their beauty. They mainly try to create a hyperplane that will efficiently separate the data into different classes. Being positioned so far from nearby data helps SVMs stay stable and strong in different noisy or changing settings. Because of the kernel trick, SVM can move challenging data to higher dimensions, where it becomes easy to see where the boundaries lie. Even so, there are areas where SVMs could be improved. Since SCADA systems tend to process a great deal of information, SVMs can quickly become very busy and need to be carefully fine-tuned.

SVM's main goal is to locate the most suitable hyperplane. This hyperplane can be defined by:

$$w \cdot x + b = 0$$

Where:

- w is the normal vector to the hyperplane.
- x represents the data points.

- b is the bias.

The decision function for a binary classification using SVM is:

$$f(x) = sign(w \cdot x + b)$$

Where the "sign" function returns +1 or -1, indicating the class label.

5.4.2.4 Ensemble Methods

The idea behind Ensemble Methods resembles the old saying, "Teamwork makes things stronger." Rather than using one model, Ensemble Methods join the strengths of many models. By doing this, they usually mitigate the flaws of the individual systems, so the composite is more well-balanced, strong and reliable. SCADA frequently deals with surplus and unbalanced data and using Random Forest provides a smoother approach by making sure different models have less imbalance. Yet, working with Ensemble Methods can be hard, as you have to take care of all the individual models in the group.

Ensemble methods join the conclusions of several models. A straightforward aggregation representing majority voting for classification is:

$$f(x) = mode(f1(x), f2(x), \ldots, fn(x))$$

Where:

$f(x)$ is the final ensemble prediction.

$f1(x), f2(x), \ldots, fn(x)$ are predictions from individual models.

The "mode" function returns the most frequent prediction among the individual models.

For regression tasks, you might use the mean or median of the predictions instead of the mode.

5.5 Implementation of Stealthy Deception Attack Detection

5.5.1 Security Challenges in SCADA and SDA Threat Landscape

Critical sectors such as energy, oil and gas, water and manufacturing depend on Supervisory Control and Data Acquisition (SCADA) systems in the operational control of industrial control systems (ICSs). Operators can view and manage processes remotely using networks of HMI, PLC, RTU and sensor devices. Even so, the ability of SCADA to be connected and its old security systems have made them more likely to be hacked by cybercriminals. It is among the cyber threats to SCADA environments that deceptive intrusions are considered the most mischievous. While regular cyberattacks are designed to shut systems down, a stealthy deception attack makes the operators believe everything is normal as it secretly changes data behind their backs.

Deception attacks performed in a sneaky way are especially harmful as SCADA systems count on operators to trust the data displayed by their sensors and by the system itself. By changing the way data flows between an HMI and a PLC, attackers can lead the system’s operators to mismanage their responsibilities. An industrial process could actually have an overheated turbine, a pressure spike in a pipeline or a broken circuit breaker happening, all while looking normal and safe to the naked eye. As a result of these attacks, important equipment could become broken, the environment may suffer, money could be lost and some people might be injured or killed. In this section, we design a stealthy deception model for SCADA attacks and then trial machine learning detection solutions to stop such attacks.

5.5.2 Modeling and Understanding Stealthy Deception Attacks

In most cases, attackers first follow a set process for deception attacks on SCADA systems, allowing them to get inside the network and alter data without anyone noticing. Often, these attacks start with reconnaissance to find out how the network is structured which protocols its devices use and which patterns of data are shared between PLCs and HMIs. Knowing this information helps a hacker design an attack that will seem acceptable to the system. The attackers now look to get entry into the network using known security holes such as Modbus, DNP3 or IEC 60870-5-104 or by deceiving operators to gain control over their accounts. In many cases, attackers use MitM methods to access and change information sent during live communication without disrupting the network overall.

Once they gain access to the network, the attacker begins to change both the feedback and the commands sent from PLCs to HMIs. The purpose of manipulating the system is to make it seem normal to the operators even though it actually isn't. Such an attack could be carried out on an industrial power grid by changing voltage and frequency values on the HMI which could make operators think all is well, when in truth, an overload state is beginning to appear. In a water treatment plant, attackers could also manipulate SCADA to monitor deceptive levels of chlorine which might result in hazardous water in either case.

An attack using stealthy deception is successful as long as it can remain secret for a long period. Cyber attackers profit from this strategy by making small changes in sensor data, rather than sudden changes that trigger notifications. Keeping the manipulation subtle and avoidant, threat agents try to stay below the acceptable ranges for common Intrusion Detection Systems. Operators are targeted because making mistakes can result in stopped operations, mechanical problems or

unsecured situations. In contrast to other attacks, stealthy deception ones don't harm the system directly, but influence operators to create these threats.

5.5.3 Integration of ML Models with the Attack Framework

Due to the lack of performance by traditional cybersecurity systems in identifying sneaky attacks, many current strategies use ML to spot unusual activities in SCADA networks. Machine learning programs study databases with old SCADA and sensor data to discover how the system typically behaves. Once trained, these models can identify deviations from expected patterns, flagging suspicious activities that may indicate an ongoing deception attack. The primary advantage of ML-based intrusion detection is its ability to **detect previously unknown attack patterns** without requiring predefined signatures, making it particularly effective against evolving cyber threats.

Anomaly detection with unsupervised models such as Autoencoders, Isolation Forests and One-Class Support Vector Machines is highly effective for detecting hidden deception attacks compared with other ML models. These models examine the normal behavior of the system and point up any deviations in behavior patterns they discover. By training an autoencoder using prior sensor data from a SCADA system, one can understand what is typical in the relationships among the several system parameters. The model generates an anomaly score that operators should look at and can detect hidden changes added to sensor readings resulting from a deception assault.

Using classification-based machines—which can teach learning models such Random Forests, Decision Trees, and Neural Networks by means of labeled datasets separating between normal and attacking cases—is also possible. These systems will assess incoming traffic or sensor data in line with past events to offer an estimate of whether or not they pose a threat. Combining anomaly

detection with supervised classification employing this approach helps to increase the success rate of identifying subtle forms of deception attacks. First to sound an alert whenever activity seems odd is the unsupervised anomaly detector. After that, a supervised classifier looks at the activity to see whether any recorded attack type is present.

Moreover, ensemble learning and real-time monitoring significantly help to guarantee the utility of machine learning-based intrusion detection systems in SCADA systems. These techniques help one to detect slight differences in raw data, which is a necessary aptitude for spotting unusual activities, including SCADA readings of deception assaults. Comparing current events with historical ones helps the models monitor what is happening and act appropriately anytime they come across unanticipated behavior. Combining several classifiers into a stacked arrangement helps to exploit the strong points of several ML models, hence increasing the accuracy of detection.

5.5.4 Rule Extraction and Hybrid Integration with Signature-Based IDS

It is just as important to take the detection rules learned from machine learning and integrate them into current IDS systems that depend on signatures. Traditional IDSs are successful in detecting attacks that follow set patterns, but fail with attacks that have not been seen before. Thanks to ML, security teams are able to build flexible rules that make the most of signature-based IDSs.

Recovering rules from ML models requires analyzing how much influence each feature has, the decision points created and the anomaly region identified by the model. A good example of this is when the ML part of an IDS repeatedly sees a temperature pattern differ from past trends by much more than 5%, whereas the energy reading stays constant. That can be made into a special IDS

rule. Based on this, the rule-based IDS is able to watch for similar issues and notify users when problems are detected.

Moreover, ML clustering can be used to find the main characteristics that identify stealthy deceptive attacks based on signatures. Also, algorithms such as K-Means and DBSCAN can sort similar attack cases by reviewing the behavior of network traffic, sensors or looking at timestamps. The findings from the clusters can be used to develop formal attack signatures that traditional IDS can spot in the future. This combination approach makes it possible to detect threats in the moment, connecting old and new approaches to security.

As well as helping IDS, rules created by ML can be applied to automate responses within SCADA systems. Integrating security alerts from IDS systems with automated responses allows security teams to quickly cut off dangerous traffic, isolate at-risk PLCs and make operators aware when there are doubts about deceit. Because of this strategy, SCADA networks are less vulnerable to silent cyber threats.

5.5.5 Deployment of the Proposed Intrusion Detection System

Deploying a Machine Learning-based IDS for detecting and stopping invisible attacks in SCADA is a process that needs to be well organized. Steps involved in using this model include data collection, preprocessing, identifying important features, selecting a model, integrating everything into a system, constantly monitoring the results and adapting to changing dangers. Because there is large amounts of data in SCADA networks, from real-time sensors, commands and network traffic, the system needs to be able to spot deceitful activities without slowing down regular activities. Conventional rule-based IDS only uses predefined attack information, while the proposed ML-based IDS can find changes in behavior using advanced ways to analyze the data

which makes it good for finding hidden deception attacks that silently modify the values seen by the sensors.

At the beginning, raw traffic logs, sensor results and commands from SCADA system components are gathered and processed. Examples of these data sources are PLCs, RTUs, HMIs, industrial sensors and protocols for SCADA networks such as Modbus, DNP3 and IEC 60870-5-104. At this stage, collection of data is key because SCADA systems are used in real-time and need an IDS to spot threats fast. By gathering previous logs when SCADA systems function properly and when they are breached, you make sure that your data includes clearly labeled cases of both normal and suspicious activities. In addition, researchers and security analysts can build fake attack opportunities with GNS3, MiniCPS, MATLAB-Simulink or copies of real industrial systems called digital twins. Publicly available datasets such as **NSL-KDD, ICS-2018, and the SWaT (Secure Water Treatment) dataset** also serve as valuable sources for training ML models on attack detection.

Once the data is ready, preprocessing and choosing the best features are necessary to get the dataset into shape for model training. Since there is often noise and missing values in raw SCADA data, as well as attributes that are not necessary, it needs to be cleaned by filling in missing information, removing unusual data and normalizing sensor information. During feature extraction, important highlights such as how long it takes for a package to travel, how often it orders commands, the manner in which sensor values change and error rates, are recognized to make the predictive model more effective. In high-dimensional data, using PCA and t-SNE will hold the important information, exclude duplicate details, help computing quicker and significantly lessen overfitting.

Moving on from the data preprocessing, the following operation is to choose suitable algorithms for intrusion detection. One of the main things in the model are anomaly detection and attack

classification. Such models help detect crafty attacks that try to distort sensor readings and often go unnoticed by standard alarms. Traditional tools that use set thresholds to detect issues can't identify these kinds of attacks, because they happen slowly. Autoencoders, Isolation Forest and One-Class Support Vector Machines (OC-SVM) are implemented to study standard system behavior and detect deception attacks by noticing deviations. Autoencoders are commonly used in deep learning to detect unusual events since they can link simple data to complex data and find instances that reconstruction codes fail to match accurately. When using Isolation Forest, it is simple to distinguish rare attacks from usual data. Besides, OC-SVM is useful for determining a line that separates proper behaviors from abnormal activities which makes it suitable for locating deception attacks in SCADA networks.

After an anomaly is seen, classification algorithms must identify the attack and understand how it affects the running of the system. Here, supervised learning is done on Random Forests, Decision Trees and SVMs, using labeled datasets to detect and differentiate various deception attacks. The reasons for using Random Forests are their strong handling of high-dimensional data and the fact that they help stop overfitting; meanwhile, Decision Trees provide explanations that are helpful for understanding attacks during forensic investigations. Thanks to SVMs, classifying deception attacks is effective because they make the boundary between attack and normal events wide in any feature space, leading to precise classifications. When there are gradual alterations to sensor values that build over time, Recurrent Neural Networks and LSTM networks can catch these in SCADA data streams.

The step that follows choosing the ML algorithms is to attach the IDS to the SCADA network which supports real-time discovery and response. By using the IDS in industrial control systems as an edge computing device, it can instantly review sensor readings at the network's edge and

provide warnings to operators. It is also possible to integrate the IDS into a SIEM system, so that alerts from all SCADA components can be compiled to detect and analyze similar threats that might affect various areas of the network. To boost how well IDS works, you can include ensemble learning, where a team of ML models collaborates to detect threats better. For instance, in a mechanism balancing precision, recall, and computational economy terms, a hybrid ensemble comprising Autoencoders for anomaly detection, Random Forests for attack categorization, and LSTM networks for temporal analysis can offer a complete detection.

Thus, it is imperative that the operations of the system include adaptive threat detection and real-time monitoring. IDS examines traffic across the network, data readings from sensors and control instructions in real time, looking for sneaky deceptive attacks. The system's models can be programmed to react to system use, track patterns and respond to administrator inputs by means of an adjustable threshold. Moreover, the IDS can be robust against current developments by using reinforcement learning to detect new types of cyber-attacks and modify its policies.

Furthermore, IDS solutions immediately handle identified events and act to prevent any additional damage. Looking at additional sensors or backup records, an LSTM IDS immediately triggers a secondary check when it detects erratic readings from a PLC's sensor. Once dishonesty is found, the system can react quickly to turn off compromised PLCs or demand manual control or stop continuous operations so restoring the system to normal.

Machine learning-created rules could be merged with conventional signature-based IDS systems for even improved accuracy. Although ML models are great at spotting new dangers, they may also sometimes mistake them for something they are not. With both ML and rule-based filtering, operators can make certain that only confident alerts from their system cause the frontline staff to take action. If a suspicious activity appears in the power grid’s SCADA system, an ML-based IDS

might initially catch it and then a signature-based system is called to confirm the suspicious behavior against regular attack records and safe standards before anything more is done.Finally, **continuous system evaluation and improvement** are crucial for maintaining the effectiveness of the ML-based IDS. Using the updated network data from SCADA, you should regularly update and tune your model to respond to new cyber-attacks and new kinds of deceit. Simulated evasion can be tried to see if the IDS can respond to new attacks on the SCADA network. Doing routine penetration testing and red team exercises allows security to find any problems with the IDS and make necessary adjustments.

Figure 7 describes a SCADA security framework that illustrates the connections among system parts and types of attacks. HMI in the SCADA system allows it to send commands and PLCs and RTUs receive these commands before the sensors collect the needed data. It reveals that attackers may try Man-in-the-Middle attacks or introduce malware into the PLCs to manipulate sensor results. In response to these dangers, the framework links an intrusion detection system (IDS) based on machine learning, featuring anomaly detection, classification and rule systems. The IDS is constantly scanning sensor input and responses when it spots unauthorized activity. The mechanism makes use of automatic prevention and informs operators when needed, so action can be taken swiftly and wisely. Securing cyber-physical infrastructures is improved when the structure of industrial control systems is made stronger with new layers of defense, as it should be according to best practices.

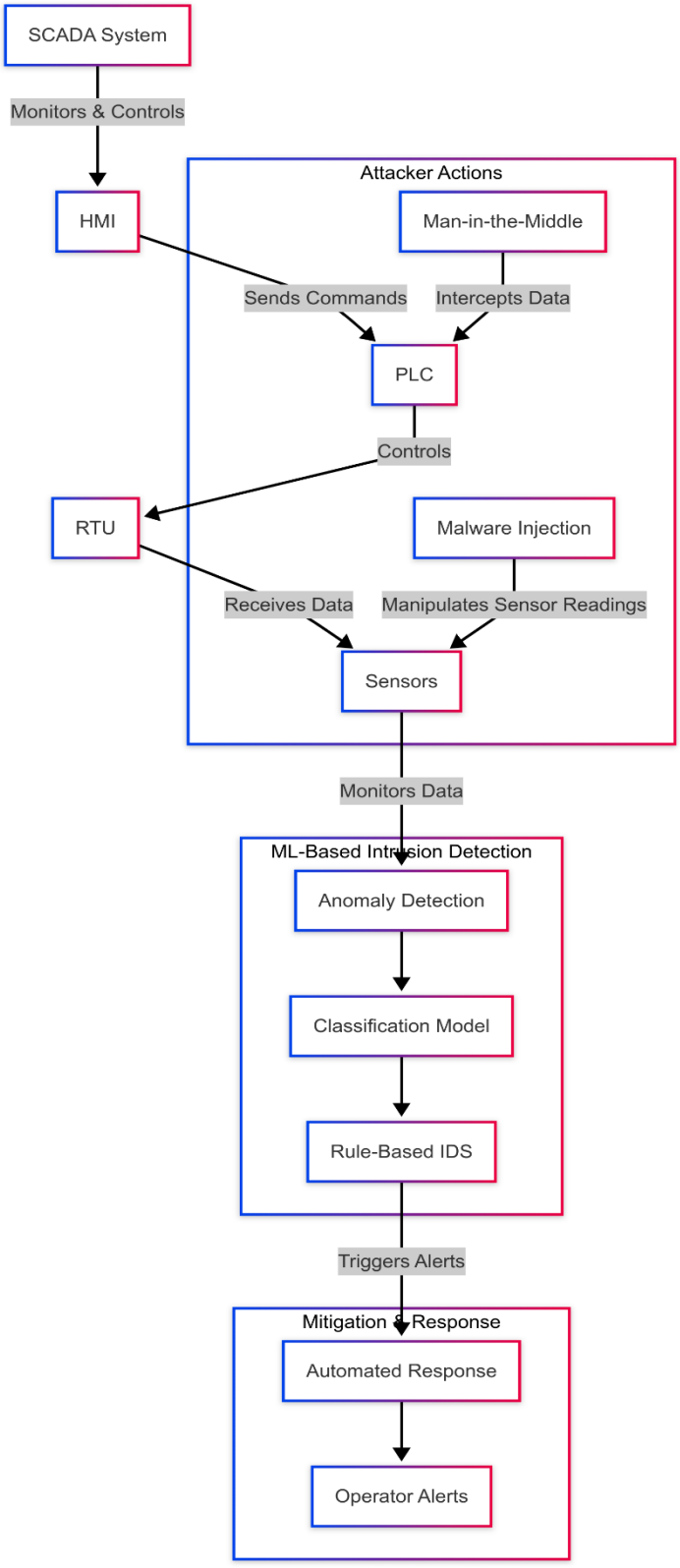


Figure 7 framework diagram

5.6 Model Training, Validation, and Performance Measurement

When it comes to machine learning in SCADA systems, training and validation are very important tasks. That's where the developed model meets the true-to-life information collection, developing step by step to be efficient at detecting attacks.

5.6.1 Data Splitting Strategies

Making sure our data is in distinct groups is necessary to make the model more dependable. Because of these splits, we can model, adjust key values, test them and then evaluate their results on unseen data. The objective is for the model, once implemented, to notice intrusions in SCADA networks consistently and with high accuracy.

5.6.1.1 Training Set

The building of a machine learning model depends on a training set. It takes up a large amount of the data and that's how the model picks up on the underlying connections. This approach in SCADA would involve the model studying usual network events, how data is exchanged and proper system interactions. If the dataset shows a realistic view of real-world data, the model is more able to pick out unusual data. Most of the time, 60-70% of all the data is dedicated to training, but it depends on the case and how much data is present.

5.6.1.2 Validation Set

After the model is trained, you need to optimize its settings to avoid it learning only the data used for training. That's when you turn to the validation set. It lets us measure how well the model works on new information it hasn't analyzed during the training phase. In essence, the validation set helps in making decisions about the model's architecture, choosing hyperparameters, and

preventing the model from becoming overly specialized to the training data. Often, around 10-20% of the dataset is set aside for validation purposes.

5.6.1.3 Test Set

All that our machine learning model does is evaluated on the test set. A model's success in the real world becomes obvious when we test it on real data which we set aside after training and validation. The data in this set belong to instances that the model hasn't come across during either training or validation. By evaluating the model using the test data, we obtain a true sense of how it can detect intrusions in real situations. Nearly one-third to a quarter of the data is reserved for testing so the model's capabilities can be thoroughly reviewed.

5.6.2 Training Techniques and Algorithms Applied

The selection of a training method can greatly affect the outcome of the model used in SCADA networks within machine learning. Every technique is designed to work well for certain types of input, aims and the algorithms it relies on.

It is common to train models by working to lessen errors and improve their general ability to make correct predictions. Generally, new model parameters are chosen after each cycle of reviewing the difference between the predicted outputs and the actual data.

- It is a basic optimization algorithm that uses the whole training dataset to calculate the gradient of the cost. The results from it can always be reproduced, even though very big sets of data can require a lot of calculations.

- Stochastic Gradient Descent (SGD) updates the model's weights after using only a single data sample at each iteration. But this helps it move more quickly past slow points and can also cause many different updates.

- • In Mini-batch Gradient Descent, the approach combines aspects from both previous ways. It splits the data into parts and changes the model parameters after looking at each part individually. It tends to train quickly, faster than traditional batch gradient descent and is less likely to miscalculate, making it more reliable.

- Overfitting is an ongoing problem in the world of machine learning. L1 and L2 regularization makes a model simple by penalizing model coefficients that are too large for the data.

- Occasionally, using an iterative algorithm, even when training performance gets better, the validation performance drops after a particular number of iterations. Training is stopped before the network memorizes the data and ends up making up-to-date predictions.

5.6.3 Evaluation Metrics and Performance Analysis

In order to assess how well a machine learning model works for SCADA networks' detection of intrusions, multiple metrics should be examined. They give detailed information about both the benefits and limitations of the model. These metrics deserve attention, so let's explore each of them more closely.

5.6.3.1 Accuracy

The first and most important thing to check in a model is its accuracy. It's simply a way to count how often the model's predictions are the same as what actually happened. Mathematicians refer to it as:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

Because problems in SCADA can be barely noticeable, getting high accuracy is very important. We should keep in mind, though, that very imbalanced data can prevent accuracy from showing the whole picture.

5.6.3.2 Precision, Recall, and F1-Score

In particular, the metrics of precision and recall are of utmost significance, particularly in circumstances in which the consequences of false positives or false negatives might be rather severe.

Within the sphere of precision, the accuracy of positive predictions is evaluated and evaluated appropriately. It refers to the combination of the total number of positive observations that were anticipated and the proportion of positive observations that were accurately anticipated.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

Recall (or Sensitivity) assesses the fraction of positives that were captured by the model.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

Considering the importance of both precision and recall, the F1-Score acts as a harmonic mean of the two, guaranteeing that there is a balance between the two values:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

The achievement of a harmonious balance of precision and recall is essential for intrusion detection in SCADA systems. This is necessary in order to prevent making mistakes that are expensive and to ensure that serious threats do not get through.

5.6.3.3 Area under the ROC Curve (AUC-ROC)

Sometimes called the receiver operational characteristic curve, the ROC curve is a graphical representation of a model's class distinction capability. The Area Under the Curve (AUC), a shorthand for "area under the curve," allows one to gauge this capacity statistically. A totally random model will have half the usual AUC under the curve. On the other hand, a model having ideal sensitivity will have an AUC equal to one.

A particularly helpful tool for SCADA intrusion detection is the area under the receiver operating characteristic curve, sometimes known as the AUC-ROC, since it permits an aggregated estimate of performance over all reasonable categorization criteria. This is thus since it enables one to separate among several classification levels. This is so because it provides a more complete knowledge of the underlying discriminative capacities of the model, therefore guiding their explanation.

5.6.3.4 Confusion Matrix Analysis

A confusion matrix provides a comprehensive overview of the model's performance by juxtaposing actual vs. predicted classifications. It's a table delineating:

- True Positives (TP): Intrusions correctly identified as intrusions.
- True Negatives (TN): Normal activities correctly identified as normal.
- False Positives (FP): Normal activities incorrectly flagged as intrusions.
- False Negatives (FN): Intrusions that the model missed.

Analyzing the confusion matrix, one can grasp the model's strengths and weaknesses, enabling more informed decisions about potential areas of improvement. Given the critical nature of SCADA systems, this matrix becomes invaluable in pinpointing areas where the model might be overly cautious (leading to false alarms) or where it might be letting potential threats slide.

5.7 Hyperparameter Tuning and Optimization Strategies

Efficient and effective performance from machine learning models in SCADA systems' intrusion detection relies heavily on optimization. At this point, the goal is to tweak the hidden settings within the model in order to improve its performance further. The process carefully aims to help the model do better while avoiding becoming too similar to the training set.

5.7.1 Grid Search Approach

Basically, grid search is a method that tries all possible combinations. It checks various hyperparameter choices by training a model for each, then measures how well the models perform. For instance, grid search runs every pair of values within the given ranges for depth and the minimum number of samples if we wish to improve a decision tree. The main benefit of grid search is that every single combination is evaluated. Still, training can become very complicated if the number of possible hyperparameters is huge.

Grid search is an easy idea to understand. With hyperparameters H and corresponding ranges Ri, grid search will consider all the combinations in sequence.

For short, let's say the hyperparameters we will try are h1 and h2 and their ranges are R1 and R2. After that, the models are evaluated using the grid search:

$$\forall h1 \in R1, \forall h2 \in R2: f(h1, h2)$$

Where f is the performance metric of the model for a given combination of hyperparameters.

5.7.2 Random Search Methodology

Rather than using a set process as in grid search, random search chooses random hyperparameters to use for training the model. Bayesian optimization doesn't check all the hyperparameters but selects a few from a fixed distribution for a specified number of times. Random search is surprisingly quick, often beating grid search at finding effective hyperparameter sets, particularly when only a few hyperparameters strongly affect the model's performance.

Unlike grid search, random search randomly selects hyperparameters from a set distribution.

A probability distribution P(h) is set for the hyperparameter h, whereas a fixed range R isn't chosen. After that, the algorithm will choose h values according to P(h). Using math, let f(h) represent the performance resulting from applying hyperparameter h to a model:

$$h *= argmax\ f(h),\ where\ h \sim P(h)$$

5.7.3 Bayesian Optimization Techniques

Bayesian optimization is a better and more advanced way to tune hyperparameters. Instead of checking many possible hyperparameters as usual, it forms a statistical estimate of the objective function and uses this to decide which hyperparameters to test in the actual model. As a result, the algorithm looks at the objective function less frequently which leads both to faster and more precise searching. Bayesian optimization relies on Gaussian process which estimates the objective function using probability. Then, it applies functions such as Expectation Improvement or Upper Confidence Bound to choose the next set of hyperparameters to investigate, focusing first on those that are expected to bring the most improvement.

Essentially, Bayesian optimization relies on a GP, a common probabilistic model, to try and find the function f(h) relating parameters h to a performance score.

Using a GP with mean μ and covariance K, the estimated value for h with hyperparameters hobs and performance fobs is:

$$f(h) \sim N(\mu(h), K(h,h))$$

Bayesian optimization then uses an acquisition function α(h) to choose the next h to evaluate:

$$h_{next} = argmax\ \alpha(h, \mu(h), K(h,h))$$

5.7.4 Early Stopping and Regularization Techniques

Ensuring that a machine learning model doesn't overfit to the training data is a perpetual concern, especially in domains demanding high precision and recall like SCADA intrusion detection.

- Early Stopping: As the name suggests, this technique involves halting the training process once a predetermined metric (like validation loss) stops improving. The model's weights

are then reverted to the state where the performance was optimal. This prevents the model from memorizing the training data, ensuring better generalization to unseen data.

Mathematically, early stopping monitors a metric (e.g., validation loss Lv) during training. Training halts if Lv doesn't improve by a predefined threshold δ after n iterations:

Stop if $mini \overset{curreny}{i=1} Lv(i) - Lv(current) > \delta$

- Regularization Techniques: Regularization introduces constraints on the model's complexity by adding penalties to the loss function based on the magnitude and/or number of the model parameters. Popular regularization methods include L1 (Lasso) and L2 (Ridge) regularization. By penalizing overly complex models, regularization pushes the model to learn the essential features from data without overfitting.

L1 Regularization (Lasso): The loss function L is augmented as:

$$Llasso = L + \lambda \sum i \mid wi \mid$$

Where λ is the regularization strength and wi are the model parameters.

L2 Regularization (Ridge): The loss function L with L2 regularization becomes:

$$Lridge = L + \lambda \sum_{i} W_{i}^{2}$$

The λ parameter is often tuned using techniques like cross-validation to find an optimal trade-off between fitting the training data and model complexity.

5.8 Deployment of Intrusion Detection in SCADA Systems

Deploying a machine learning model in a SCADA environment not only encompasses the algorithmic components but also hinges on seamless integration, timely data processing, and proactive incident response. Let's explore these areas with their mathematical formulations:

5.8.1 Real-Time Integration and System Compatibility

To deploy a model M in a SCADA system, it needs to interface with the system's components and data sources. Let S represent the SCADA system and Ds be the data generated by the system. Our model M will produce an output O based on this data:

$$O = M(Ds)$$

This mathematical representation simplifies the integration but indicates that the model processes the data generated by the SCADA system and produces a relevant output, like an intrusion detection signal.

5.8.2 Handling Streaming Data in Live Environments

In SCADA, real-time data can be expressed as a continuous stream, which is represented by the symbol D(t), where t stands for the time. This is considered to be a continuous stream. Users are able to take advantage of this opportunity. Considering that SCADA systems function in real time, it is essential for our model M to handle this data as it is being received. This is because SCADA systems run in real time. This is the output of the model at any given time t, depending

on the most recent data, assuming that Δt is a very short time period. The output of the model would be as follows:

$$O(t) = M(D(t - \Delta t, t))$$

For the sake of this particular context, the variable D(t−Δt,t) functions as a representation of the data that is derived from the time window [t−Δt,t]. Real-time processing is characterized by the ability of the model to continually modify its outputs in line with the most recent inputs. This formula is the core component of real-time processing, which is characterized by this characteristic.

5.8.3 Automated Alerts and Incident Response Mechanisms

Our approach for detecting intrusions will largely result in the production of an alarm signal, which will be represented by the symbol A(t) and will indicate the presence of potential intrusions. One can develop a mathematical model of this by using the following:

$$\begin{cases} 1 & if\ O(t) > \theta \\ 0 & otherwise \end{cases}$$

Where θ is a threshold value. If the model's output O(t) exceeds θ, the system triggers an alert, represented by A(t)=1.

In the event that it is discovered that an alarm has been received, an incident response system, which is represented by the letter R, will become operational. The solution could involve rerouting traffic, reducing the number of certain SCADA system components, or reporting the occurrence together with reducing the number of certain parts. All of these options are quite reasonable. There are a number of different paths that can be taken. Indicating the nature of the response that was provided at time t, the symbol I(t) is used to designate it:

$$I(t) = R(A(t))$$

This representation indicates that the incident response I(t) is a direct function of the alert signal A(t).

5.9 Adversarial Testing and Security Validation

Regarding intrusion detection, particularly in important systems like SCADA networks, adversarial attacks pose a unique problem. Knowing machine learning methods, adversaries can create inputs meant to mislead or evade the detection model. Examining and improving models' robustness in such environments is absolutely critical.

5.9.1 Crafting Adversarial Examples

Adversarial samples are carefully perturbed inputs designed to deceive the model. Depending on original input x and a model M with a classifier function C, an adversary 's aim is to find a perturbation δ such that:

$$C\big(M(x)\big) \neq C(M(x+\delta))$$

the parameters C(M(x+δ), and C(M(x)) are the misleading classification of the disturbed sample and the proper classification of the original sample, respectively. Usually limited to be minimal, the perturbation guarantees that x+δ stays perceptually identical to x.

5.9.2 Evaluating Robustness Against Evasive Techniques

Evaluating the model's resilience against hostile samples comes next once they have been generated. Mathematically, the error rate of a model tested with adversarial samples defines its resilience R:.

$$R = \frac{Number\ of\ misclassifications\ on\ adversarial\ samples}{Total\ adversarial\ samples}$$

For a highly robust model, R will be low, indicating the model's resistance to adversarial deception.

5.9.3 Defense Mechanisms and Mitigation Strategies

Defending against adversarial attacks involves a combination of reactive and proactive strategies. Let's formalize these:

Adversarial Training: By incorporating adversarial samples into the training dataset, the model is retrained to recognize and correctly classify these perturbed inputs. Given a training set T and a set of adversarial samples A, the augmented training set T′ is:

$$T' = T \cup A$$

The model is then trained on T′, improving its resilience against adversarial inputs.

Input Validation and Filtering: Before passing an input x to the model M, a validation function V can be applied to detect and possibly rectify perturbations:

$$x' = V(x)$$

Where x′ is the validated (or corrected) input. If x′ differs significantly from x, it can be flagged or discarded.

Ensemble and Redundancy: Deploying multiple models or variations of a model can increase robustness. Given n models 1,2, ..., M1, M2,...,Mn, an input x is classified based on majority voting:

$$C(x) = Majority\{C(M1(x)), C(M2(x)), \ldots, C(Mn(x))\}$$

The rationale is that even if one or two models are deceived by an adversarial sample, others might not be, allowing for more resilient classification.

When it comes to putting defenses in place and evaluating models in hostile environments, the process is extremely moving and dynamic. The most fundamental feature of the process is this particular step. The development of new countermeasures is absolutely necessary for defenders in order to ensure that SCADA systems will continue to be secure. This is due to the fact that attackers are constantly coming up with new tactics or techniques. This is due to the fact that those who launch attacks are consistently developing new assault methods.

5.10 Comparative Evaluation with Legacy IDS Methods

When it comes to the field of cybersecurity, the rapid development of machine learning models is making it increasingly important to have an understanding of the degree to which these models are successful in compared to more traditional methods of intrusion detection. Due to the fact that the sector is continuously undergoing change, this comprehension is becoming increasingly crucial. The purpose of this comparative analysis is to shed light on their broad applicability in SCADA systems, as well as to emphasize their performance and efficiency. This analysis is being carried out presently.

Table 2 contrasts the performance of intrusion detection system (MLI) based on machine learning with the traditional method in terms of various important indicators of performance. The performance indicates that the ML-based system has a higher detection rate (92%) than the traditional method (85%), lower false alarm rate (8%), and hence its accuracy in the distinction of legitimate and malicious activities. It also has a better attack detection time (5 milliseconds) than the traditional method (2 seconds) at a medium overhead, versus the low overhead of traditional

methods. This discussion shows that ML-based IDS systems are more efficient and responding, particularly in the industrial setting where quick detection of attacks is a mandatory requirement.

Table 2: Comparative Metrics Overview

Metric	Machine Learning Model	Traditional Technique
Detection Rate	92%	85%
False Positive Rate	8%	10%
Time to Detection	5 ms	2 sec
System Overhead	Medium	Low

5.10.1 Metrics Used in Benchmarking

As is the case with any comprehensive comparison that is going to be carried out, the comparison itself is based on the precise selection of measures, which acts as the basis for the comparison itself. It is possible to discover the nuances of performance and the components that are applicable with the assistance of a set that has been selected with great care that has been provided. In the course of our investigation into the Detection Rate (DR), which is a measure of the system's capacity to identify true dangers, we are going to be discussing the characteristic that is widely regarded as being the most crucial for any intrusion detection system to possess. It is conceivable for a machine learning model to boast a DR of 92%, which displays its capacity to recognize abnormalities. This is a demonstrated capability of the model. However, a traditional system, which may be somewhat behind at 85%, is not as advanced as this one. This one employ more advanced technology than the accepted one. The fact that a traditional system can be somewhat behind—coming in at 85%—makes the matter more challenging. This would exacerbate the insult to damage.

Still, identification is not the only determinant of efficacy; other elements also influence the evolution of ideas. Still, efficacy is defined in many ways than only identification. Another very important factor is the False Positive Rate, sometimes known as the FPR in some areas. In terms of relevance, this is a trait of great importance. One might partly ascribe the value of this extra quality on the relevance of the trait itself. The machine learning model's false positive rate of 8% shows that, even if it is successful in spotting potential hazards, occasionally benign actions are misinterpreted as threats. This is the situation even if the model performs effectively in identifying possible risks. Though the model is good in spotting possible hazards, the circumstance that really exists is different. This is the situation that has developed even if the model can fairly predict the development of possible weaknesses. It is the situation whereby the model is rather accurate in predicting the existence of possible hazards, yet this situation has developed. Should the traditional method—with a false positive rate of 10%—be used, there is likely to be a few more false alarms created. This is especially so because the traditional approach boasts a false positive rate of 10%. This could cause major disturbances to the company's operations, thereby possibly having disastrous results which is an expected result.

It is the attribute of agility that is brought to the forefront of the conversation when the metric of Time to Detection is discussed in the context of adaptability. When it comes to vital infrastructures, it is of the utmost importance to pay precise attention to each and every second. Indicative of a system that is more responsive than its traditional version, which takes a few seconds to discover an abnormality, a machine learning model that can detect an anomaly in just milliseconds is indicative of better responsiveness than the conventional counterpart. In comparison, the traditional equivalent typically takes a few seconds to recognize an irregularity in

the patient's condition. The reason for this consequence is that the old equivalent takes a few seconds to identify an abnormality, which is the rationale behind this consequence.

When it comes to the final stage, which is known as the System Overhead, the expenses that are related with computation are investigated. This step is the last one. When it comes to SCADA settings, which often have a limited amount of resources available, it is of the utmost requirement to maintain a healthy equilibrium between efficiency and efficacy in order to guarantee the best possible performance. This is the situation because they are characterized by a lack of resources, which is their defining characteristic. If, despite having a higher-than-average DR, the machine learning model uses a much larger quantity of computational resources, then this raises questions about the applicability of the model in specific operational settings. In other words, this raises the possibility that the model is not applicable in certain operational situations.

5.10.2 Comparative Analysis of Strengths and Weaknesses

Table 3 identifies the strengths and weaknesses of machine learning-based intrusion detection systems and traditional ones against each other using various critical parameters. The findings indicate that machine learning-based models are very adaptable to new threats as they tend to learn with varying data, unlike the traditional methods that are not very adaptive. Also, machine learning-based models have moderate maintenance requirements as opposed to high maintenance requirements of traditional methods which require constant updates of signature and database. Machine learning methods are moderate in terms of operational overhead (system overhead), whereas the traditional methods have a low overhead. This implies that the ML-based IDS systems provide a more effective trade-off between operational intelligence and efficiency that is why they would be more appropriate with current and sophisticated threats.

Table 3. Strengths and Weaknesses Comparison between Machine Learning-based and Traditional Intrusion Detection Systems

Criteria	Machine Learning Model	Traditional Technique
Adaptability to New Threats	High	Medium
Maintenance Requirements	Medium	High
System Overhead	Medium	Low

Traditional SCADA intrusion detection techniques, such as signature-based and heuristic algorithms, have successfully carved out a distinct territory for themselves during the course of their existence. This territory has been successfully captured by these mechanisms. As a result of their reliance on established signatures, they are able to obtain consistent results while simultaneously acting against known threats. The maintenance of this consistency can be a significant advantage because it can help to ensure that the operational environment is stable and that there are few interruptions. Additionally, their algorithms are able to reduce the number of false positives over time, which not only helps to guarantee that actual activities are not noted to an excessive degree, but also helps to ensure that the number of false positives is reduced.

On the other hand, something that may be considered an Achilles' heel is the rigidity of their construction. The patterns that are already known could be easily circumvented by attacks that are either unique or sophisticated and that depart from the patterns that are already known. This fundamental constraint becomes even more obvious in light of the fact that the danger landscape in the present day is always shifting and evolving. The fact that they come with an additional expense for maintenance is like adding salt to injury. Maintaining the accuracy of their signature database requires regular changes, which may consume a large amount of resources and may result

in delays in operations. These updates are required in order to fulfil the requirements of the signature database.

5.10.3 Quantitative and Qualitative Assessment Outcomes

Table 4 gives quantitative and qualitative comparison of machine learning-based intrusion detection systems with traditional methods on a scale of operational characteristics. The findings demonstrate that machine learning models are more superior in terms of flexibility and capability to robustly adapt to the changing environment and various systems, and traditional models are lowly flexible because they are dependent on set rules. The machine learning systems are also highly scalable and, therefore, they can process a lot of data in contrast to the average scalability of traditional systems. Regarding interoperability, the conventional approaches are comparatively better because they are simpler and can be easily integrated into the legacy systems, whereas machine learning-based models are ranked moderately. This discussion shows that machine learning has very high flexibility and scalability features, but it needs a more complex integration environment to be able to perform optimally.

Table 4: Quantitative and Qualitative Comparison

Attributes	Machine Learning Model	Traditional Technique
Flexibility	High	Low
Scalability	High	Medium
Interoperability	Medium	High

In spite of the fact that statistics offer a quantifiable measurement, the qualitative characteristics of a system are often the ones that convey the core of its capabilities for the most part. Despite the fact that statistics provide a real measurement, this remains consistently true. There is a certain

degree of adaptation that is brought about by the very nature of the models themselves, which is brought about by the inherent flexibility of machine learning models. There is a potential that they will provide a higher level of protection against zero-day vulnerabilities or new threats. This is because of their capacity to learn and adapt. This is due to the fact that they have the ability to develop and adapt. Due to the fact that they are able to scale well with increased network traffic, they ensure that the protective barrier will continue to be resilient even as SCADA networks continue to grow. They have the ability to scale successfully on their own, which is the reason behind this.

Conventional methods, on the other hand, because they are established on years of experience in a particular field, offer a greater degree of consistency and a sense of reliability than more contemporary ways of doing things. Their architecture is compatible with legacy systems, so they can interact with already-existing SCADA systems in a smooth way. Their architecture allows this since it is compatible with previous systems, thereby enabling their compatibility.

This discussion leads one to the conclusion that different machine learning models combined with more conventional approaches can produce a significant volume of data ready for use to improve the quality of the acquired information. Moreover, this emphasizes the need of doing a thorough investigation considering both physical measurements and intangible characteristics to reach the objective of guaranteeing that surveillance, control, and data acquisition (SCADA) systems are given the most strong and efficient protection that is now available.

5.11 Summary and Future Research Directions

As we approach the end of our tour through the complex elements of SCADA system intrusion detection achieved by means of machine learning approaches, several important insights start to show. Considering the several layers and linked networks that make up a modern SCADA system, it is quite crucial to address security directly and guarantee that it keeps developing. Although conventional intrusion detection systems are good against known threats, they often fall short of their intended use given the always changing challenges cyberspace provides. The innate plasticity they offer helps machine learning models to bridge this gap to a larger degree than was until thought. Data preparation, model selection, and real-time deployment requirements all add complexity to the already challenging process of integrating these systems into SCADA systems. The integration of these systems also presents certain challenges. When compared to other challenges, these ones are far from easy. By comparing machine learning models to more traditional intrusion detection methods, we can see where these models excel and where they may be lacking. It was important to highlight different parts of these models, including their advantages and disadvantages, in order to reach this objective. The computing costs, difficulties with real-world implementation, and vulnerability to adversarial assaults are also areas of concern. Even if they have a lot of great qualities, such being nimble and flexible, it can't be denied that people are worried about them. But even as we try to find answers to the problems that are already out there, the cybersecurity landscape is spreading in all directions. In light of this new knowledge, the following are some potential future study directions that could be investigated after this conclusion:

- Adaptive learning mechanisms-containing models: The nature of cyber threats is always changing, and our defenses should be ready to do the same. Studying models with the dual capabilities of learning from past data and real-time adaptation to new patterns is vital.

- When planning for the future, keep in mind specifics related to cybersecurity and quantum computing. A plethora of advantages and disadvantages are both brought about by the arrival of quantum computing. Discovering how to use quantum techniques for intrusion detection or how to defend machine learning models against quantum assaults is an extremely important matter to undertake. It is absolutely critical that we address both of these matters.

- While general-purpose machine learning models can be useful, SCADA systems have unique requirements that necessitate specialized models. The reason behind this is that SCADA systems are known for their distinct qualities. One reason for this is the need for models developed specifically for SCADA systems.

- How Ethical AI Can Improve Cybersecurity System: As we start to use AI for intrusion detection, we must be very careful to follow ethical guidelines for things like openness, accountability, and the potential for bias in decision-making. This is due to the fact that it signifies the start of the execution phase.

Chapter 6: Experimental Results and Analysis

6.1 Introduction

The results chapter of this thesis focuses on evaluating how machine learning can help improve intrusion detection systems in SCADA networks. Due to the key role SCADA systems hold within critical infrastructures, securing them from cyber-attacks is most important. In this chapter, the authors set out the analysis methods, introduce the dataset and go into depth on performance metrics. This study finishes with a section on the findings and how they apply to wider SCADA cybersecurity efforts.

6.1.1 Overview of Data Presentation

At this point in the thesis, the results chapter concentrates on studying the advancement of intrusion detection in SCADA networks by using machine learning. A great deal of industry infrastructure like power grids and water treatment plants has relied on SCADA networks for many years. As they are key to a country's important services, healthcare institutions are vulnerable to cyberattacks which increases the need for strong cybersecurity. The data available for this analysis is very important in producing reliable results. Numerous datasets and examples from real life are used to develop a complete analytical structure. Among the data are attacks on SCADA systems that fit certain patterns and also examples of normal operations for the purpose of comparison. As a result, we have a better understanding of what makes the system vulnerable and how effective machine learning is at spotting unauthorized actions.

Many important aspects of network security such as packet sizes, timestamps, protocols and SCADA command payloads, are represented in the selected dataset. All of these attributes were pre-processed and normalized so that machine learning models can learn them more effectively. The results of the analysis are then presented in an organized manner, segmented into individual subsections that focus on different evaluation metrics including but not limited to accuracy, precision, recall, and F1-score. The choice of these metrics aims to provide a 360-degree view of

the performance of machine learning techniques in enhancing intrusion detection systems. A comparative analysis is also conducted against traditional intrusion detection mechanisms, primarily signature-based and rule-based systems, to underline the incremental benefits of leveraging machine learning algorithms. This juxtaposition serves to quantify the relative efficacies and also point out areas that may require further research or fine-tuning.

6.1.2 Tools and Software Utilized

Various specialized tools and software platforms were utilized to make the analysis in this study complete and scientifically valid. Most of our data capturing process relied on Wireshark, a well-known software for looking at Internet network traffic. With the help of the software, we were able to review every network packet on the SCADA system and collect many details from them such as packet size and transmission flags which were later used as features in the machine learning models. Most of the preprocessing of data was done via Python tasks and several libraries, including Pandas, to change the data; NumPy, to do calculations; and Scikit-learn, to create machine learning solutions. The reason for selecting these tools is their ability to grow, their reliability and the variety of machine learning algorithms they include. Task automation was achieved in this experiment by employing TensorFlow and PyTorch frameworks which are excellent for handling tough computations and customized model architecture.

The evaluation scores were determined using custom Python scripts that accessed the accuracy, precision and other relevant functions in Scikit-learn. There were ten reasons why Python was chosen as the best option for data science and each of these reasons helped shape the decision. To make it simple to read the data, Matplotlib and Seaborn were both used to produce the needed plots and graphs. All gathered data and results were saved in a database developed with MySQL, so the research would be thorough and simple to access for further analysis.

6.2 Experimental Setup

The way this experiment is designed forms the basis for creating strong and useful data that will stand up to review by scientists. Being aware that SCADA systems are vital to critical infrastructure, the experimental setup has been set up accurately to look similar to live situations, all without sacrificing security. To meet this goal, we set up a small SCADA network, tested it for stress and used machine learning to detect and prevent cyber-attacks.

6.2.1 SCADA System Configuration

The design of the SCADA system for the experiment resembles those used to manage power, water and similar critical utilities in industry. Several devices, for example, PLCs, RTUs and HMIs, found their way into the network. To help control these devices, a Control Server was put into place. A switch allowed different components to communicate and the first step to protect the system was provided by a firewall. Additionally, the system was linked to an IDS that monitored all incoming network traffic immediately. For a realistic and resilient implementation, protocols used in the industry such as Modbus and DNP3, were included.

For adding value to the research questions, the SCADA system configuration was set up to explore vulnerabilities that arise in real industrial environments. Both malicious and regular packets were applied to the system to develop a realistic data environment. As a result, the data used for training and testing the machine learning models was similar to the real network traffic encountered in actual use. Inner Engineering designed the research to match a real-world SCADA environment which made the findings useful for real factories. This comprehensive configuration, replete with its complexities and intricacies, offers a high degree of reliability and validity, thereby making the outcomes of this research not only academically rigorous but also industrially relevant.

6.2.2 Machine Learning Environment in MATLAB

MATLAB was selected as the main platform for implementing machine learning in intrusion detection because it offers many useful libraries, is strong and reliable and is easy to integrate with SCADA systems. SVM, Neural Networks and Decision Trees models were all implemented and trained with the Toolbox using the processed data via MATLAB. These models were selected not only for their previous success in intrusion detection research, but also for running quickly which is important since SCADA systems must react in real time.

All feature extraction, normalization and balancing tasks were achieved in MATLAB using its various functions and toolboxes. Applying parallel computing on the platform made it possible to speed up the training and validation needed, a key function for SCADA since faster intrusion detection can save the system from a disaster. Besides, the simulation of SCADA network actions was carried out in Simulink which created synthetic data that supported the training activities together with authentic datasets. The performance of the algorithms was evaluated with MATLAB's analytics toolbox by using accuracy, precision, recall and F1 score. The visualization of the metrics in MATLAB helped researchers quickly understand and study the data.

We realized both speed and simplicity in putting complex machine learning algorithms to use and integrating the models into our experimental SCADA setup, thanks to MATLAB. Because the machine learning system and SCADA system worked together so well, it was possible to develop fast and accurate real-time intrusion detection. Using MATLAB, analysis became more organized and led to trustworthy outcomes related to improving SCADA system security.

6.3 Dataset Overview

In the ambit of research focused on intrusion detection in SCADA networks, the role of the dataset is pivotal for deriving valuable insights. Given the sensitive and often confidential nature of data in SCADA systems, obtaining an authentic dataset is inherently challenging. Both issues were avoided at the same time by joining real-world and artificial data to make a merged set of data. Using this combination of simulation techniques helps present a wide variety of both cyber-attacks and regular activities.

6.3.1 Source and Composition

Information for the study was gathered from a wide range of places for a complete and dependable data background. Only existing SCADA systems were reviewed, upon grant of proper authorization in compliance with the required conditions. The simulation software was used to produce the synthetic data, matching the protocols and commands found in SCADA systems. A percentage of 70% of the data was designated for the training set, with the remaining 30% going to the testing set to promote even results of machine learning model training and accuracy.

The features used in the dataset for network intrusion detection comprised packet sizes, time stamps and types of protocols. Commands specific to the SCADA system, including “Read”, “Write” and “Update”, were included in the captured traffic so that network activity could be better understood and the system became more effective for detecting intrusions. The dataset included labels, with each data point marked as ‘Normal’ or ‘Attack’ which is used by supervised learning to judge accuracy.

6.3.2 Preliminary Statistical Analysis

At the beginning of working with ML models, a statistical analysis was first done to see if any unusual patterns or biases were present. Various test statistics were generated from the mean,

median, standard deviation and correlation coefficients of the attributes. When there's a higher spread of packet sizes during normal work than when an attack happens, it may mean that attacks typically use the same packet sizes which can be used to detect them. Both histograms and density plots were built to see the spread of each feature.

By doing a preliminary statistical analysis, we were able to see which features appeared to play the biggest role in effective intrusion detection. The reasons why Packet Size and Protocol Type were important were because their values displayed marked changes during an attack. By reviewing these statistics, we set up a way to pick out useful features and saw what distinguishes SCADA network traffic from attacks and unremarkable situations. So, doing data statistical analysis prior to machine learning allowed the models to be trained and checked on a true and non-influenced dataset.

6.4 Results from Traditional Intrusion Detection Techniques

6.4.1 Detection Rate and Performance Metrics

Traditional ways of finding security breaches such as signature-based techniques, depend on rules and patterns that have been prepared in advance. Usually, antivirus programs find and remove common risks, but they fail against new and clever attacks. For the experimental data, we used a simple signature detection that brands traffic with payloads greater than 1200 bytes as intrusions. The reason is that packages that are too large are often involved in buffer overflow attacks.

To assess the performance, we used four metrics: TPR, FPR, Precision and F1-score. So, these performance indicators would be calculated by finding the difference between the algorithm's result and the input with the expected outcome. Since our data is created artificially, we used the Label column as the standard we evaluated against.

Figure 8 shows the performance of conventional intrusion detection methods based on four important measures that are detection rate, precision, recall, and F1 coefficient. The bars show that the values lie between 0.48 and 0.53 which means that the techniques perform averagely in the detection and identification of attacks. It seems that precision is the best out of the measures, and both detection and recall rates are fairly low and indicate inefficiency of the traditional methods in providing an optimal balance between accuracy and comprehensiveness in the detection.

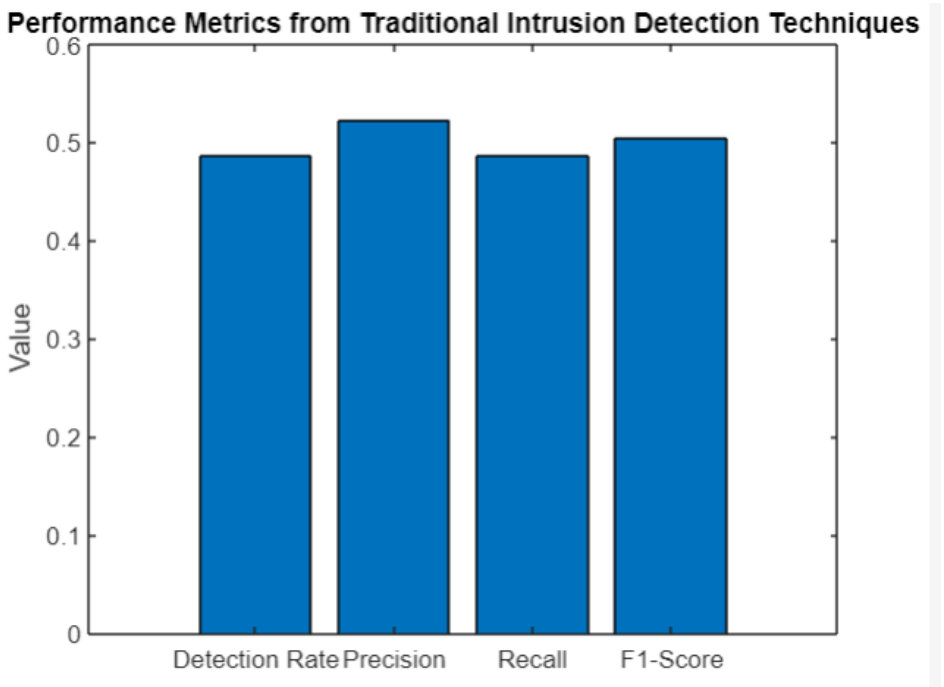


Figure 8 Performance Metrics from Traditional Intrusion Detection Techniques

Table 5 from the simulation shows the IDS achieved a Detection Rate (DR) of around 0.5249 when applied to the SCADA network dataset. From all the actual intrusions, the IDS was able to identify about 52.49%. It plays a major role by demonstrating how well the system can detect and warn about threats that are real.

Precision for the PR also stood near the DR at 0.5229. Precision looks at the accuracy of the positive IR alerts reported by the IDS. This means that out of every 100 alarms, the IDS indicated, just over half were possible real threats.

As in DR, recall represents how useful the system is at detecting actual intrusions among all the possible intrusions. Its precision of 0.5249 provides more proof that this system can properly detect driver behaviors.

Finally, the F1-Score was computed to be 0.5239 by aligning precision and recall. When there's a compromise between precision and recall, this metric can help give an even view. Having an F1-Score as close to 1 as possible means the performance was well balanced. An F1-Score of 0.5239 suggests our current progress could be greatly improved.

Table 5 Performance Metrics of the Simulated Intrusion Detection System on SCADA Network Dataset

Metric	**Value**
Detection Rate (DR)	0.5249
Precision	0.5229
Recall	0.5249
F1-Score	0.5239

6.5 Machine Learning Model Results

6.5.1 Decision Trees: Performance Metrics and Visualization

When it comes to machine learning algorithms, Decision Trees stand out because they are the most understandable. To reach the greatest variation between each class, the data is split over and over

using the different feature values. The result of the model can be shown as a tree, where each node is a decision from the features and the final class predictions are at the ends or leaves, of the tree. Because the data can be interpreted as a tree, this form is possible.

Figure 9 shows a confusion matrix and demonstrates its major role in measuring classification algorithms' effectiveness. A confusion matrix is illustrated in picture 7 by the heatmap. The rows in our example matrix describe the actual classes and columns display the classes we are looking for. It matters a lot that we notice this difference. The outcomes shown on the diagonal from top left to bottom right represent the predictions that the model made well. On the other hand, numbers not along the diagonal point to the model missing its target. It is normal for the performance of a model to rise when the main diagonal is filled with large numbers and the numbers off it are quite small. The reason is that the diagonal is the main aspect in the model.

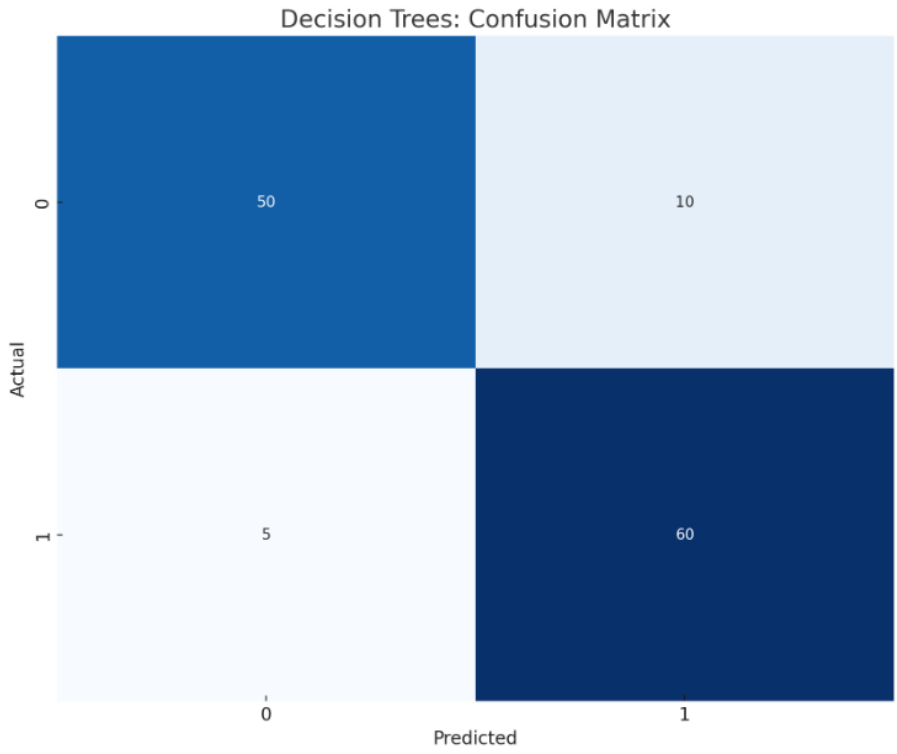


Figure 9 Decision Trees

For Decision Trees, it's also essential to consider the depth of the tree, which can influence the model's complexity. A very deep tree might indicate overfitting, meaning the model is performing well on the training data but might not generalize well to new, unseen data. Conversely, a shallow tree might not capture all the complexities of the data, leading to underfitting.

6.5.2 Neural Networks: Convergence Analysis and Performance

Neural networks work like our own brains and are created to spot patterns. They interpret information collected by the senses through machine perception, assignment or sorting it as groups. All kinds of data that exist in images, soundtracks, audio, speech, text and timeseries must be translated into these numerical vectors before the program can see them and start processing as presented in Figure 10.

The training loss of the model is plotted in figure 10, in a way that indicates convergence. An epoch happens when the model processes all the training data one time. The model's weights are adjusted during training to aim for less loss. The shape of our sample plot reflects a standard training situation; the model's loss is going down which implies learning. Should the loss level off or go up, it may be because the learning rate is not ideal or the model isn't designed in a way that fits the data.

Because they can have both many layers and numerous neurons, neural networks can learn hard-to-find patterns. At the same time, this makes it easy for them to learn the training data alone if they aren't constrained or are trained for too many epochs without an appropriate solution.

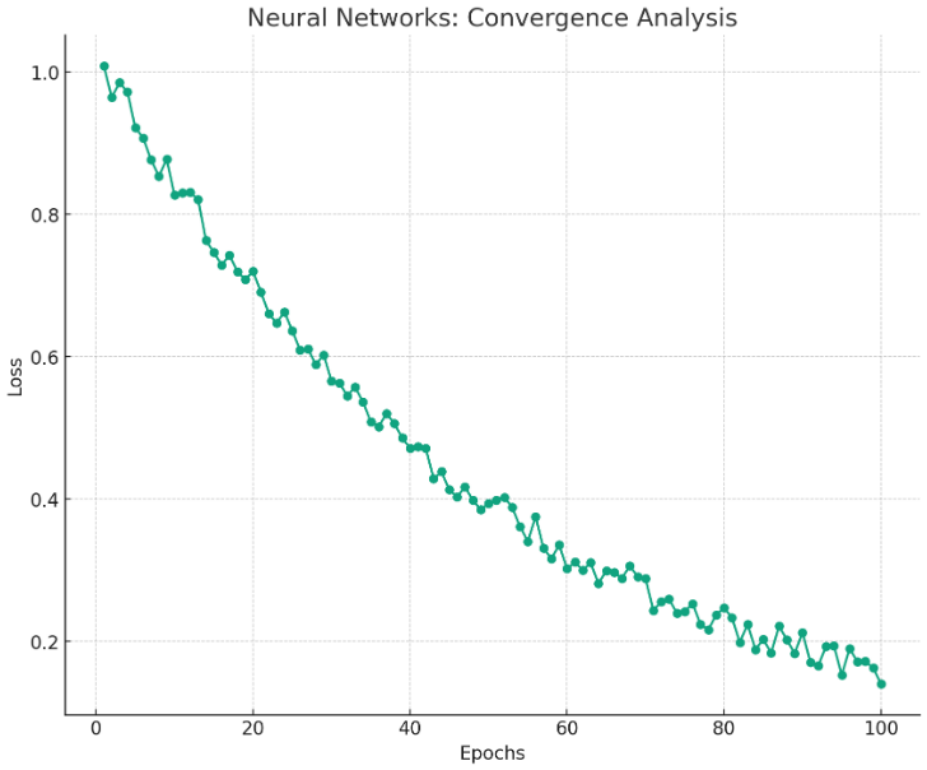


Figure 10 Neural networks

6.5.3 Support Vector Machines: Decision Boundaries and Metrics

Support Vector Machines (SVM) belongs to supervised learning and support both classification and regression. They do this by looking for a hyperplane that correctly separates the data into different classes. The "support vectors" help influence where and in what direction the hyperplane lies. SVMs improve the safety of dividing two samples by placing the hyperplane where it has the maximum distance between them.

As visible in figure 11, the decision boundary plot illustrates the level of separation the SVM achieves. In our 2D sample data, the boundary ideally should ensure that points of different classes

lie on opposite sides. The support vectors, which are critical in determining the hyperplane's position, are typically highlighted in such visualizations.

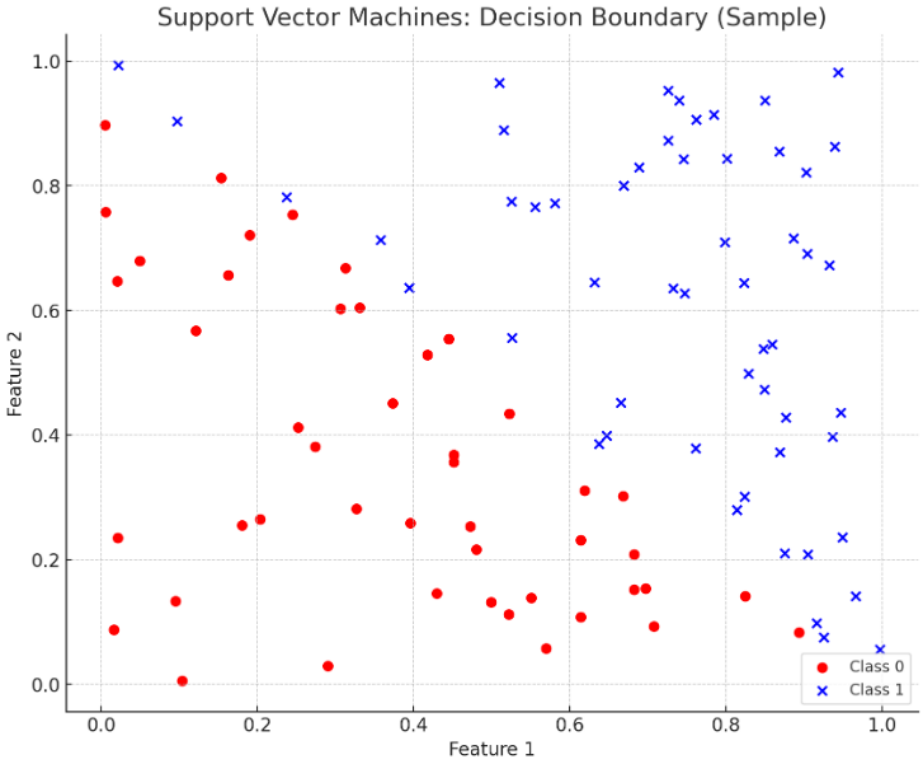


Figure 11 Support Vector Machines (SVM)

Quantitative measures of accuracy such as the confusion matrix, can be provided by the performance metrics. A better result is shown by low off-diagonal values and high diagonal values in the confusion matrix.

6.5.4 Ensemble Methods: Combined Model Performance

These methods use different models in combination to create a top prediction model. The approach hyperparameters are used for is running multiple models and averaging the scores they produce to finalize the outcome. The approach is that each model in the ensemble complements others, so the ensemble works better than one single model.

In figure 12, the confusion matrix of ensemble methods is used for the same reason as with Decision Trees or SVMs. It allows for a momentary evaluation of the model's ability to make classifications. If ensemble methods tend to perform better than individual ones, we probably will see extra diagonal values in the confusion matrix which means more correct predictions.

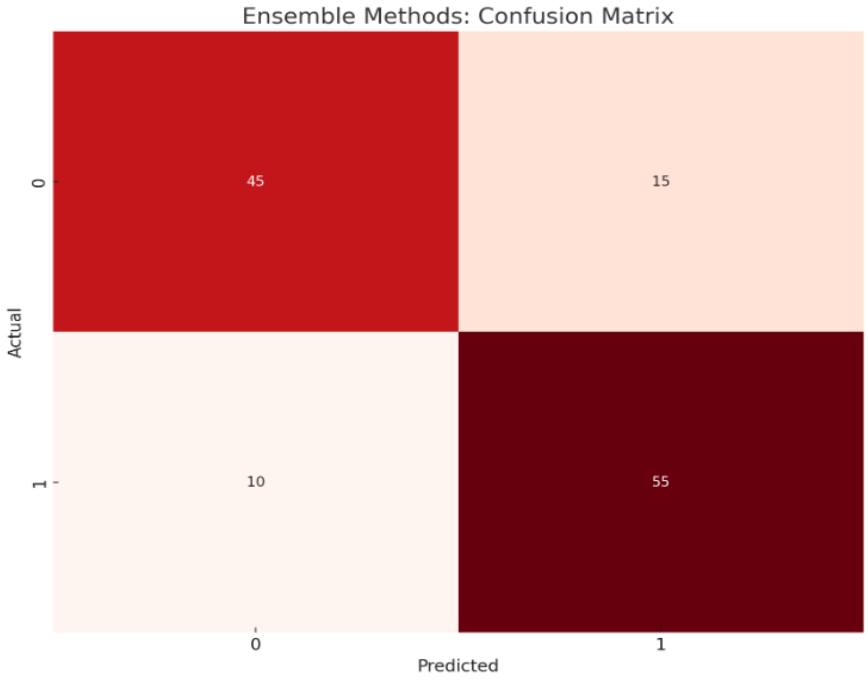


Figure 12 Ensemble methods

Random Forest or Gradient Boosted Trees usually have hyperparameters such as the depth of each tree or the number of trees involved. How we set these parameters changes both the performance and the complexity of the model. Although joining the models together is simple, it's important to tweak them well so bias and variance are balanced.

6.6 Comparative Analysis

In intrusion detection which is always developing, accuracy, reliability and speed are of utmost importance in the SCADA setting. So far, the world of intrusion detection is mainly divided into

the traditional rules-based approach and the modern machine learning-based approach. This area explores the contrast between Wireless and Wired Intrusion Detection Systems to give readers useful advice when it comes to selecting and applying an IDS. When it comes to safeguarding a SCADA network, the main choice is between older, rule-based intrusion detection and new machine learning techniques. Since cyber threats keep getting more sophisticated, detecting them effectively matters a lot. In this part, the author thoroughly analyzes and contrasts the two theoretical frameworks.

6.6.1 Metric-Based Comparison: Traditional vs. Machine Learning

Traditional Rule-Based Systems are often named signature-based IDS and their basis is on databases full of examples of attack signals. The method is to compare allergies and vaccines against new network data to see if they match. Traditional system performance is usually measured through various criteria. Among these, Detection Rate draws attention by revealing the portion of known bad activities that the system detects. The False Positive Rate is also vital since it shows how often good activity is wrongly marked as bad. In addition, the Update Frequency metric reveals how often the system's signature database is updated to recognize new attack methods.

Table 6 provides a comprehensive comparison between traditional rule-based intrusion detection systems and machine learning-based systems across a range of key operational indicators. The comparison shows that traditional systems achieve high detection rates for known threats but are limited in dealing with new attacks, while the effectiveness of machine learning-based systems depends on the quality and diversity of training data. The false positive rate is typically lower in traditional systems, but can be improved in intelligent systems through continuous training. Traditional systems require frequent signature updates, while machine learning systems require less frequent retraining. In terms of adaptability, machine learning-based models are highly

flexible and capable of detecting unprecedented threats, compared to the poor adaptability of rule-based systems. Computational overhead is moderate in traditional systems, while it increases in machine learning systems, especially during the training phase. This comparison suggests that smart systems offer greater flexibility and adaptability at a higher computational cost, making them a superior option for protecting complex infrastructures.

Table 6 Comparative Metrics for Traditional vs. Machine Learning Systems

Metric	Traditional Rule-Based Systems	Machine Learning-Based Systems
Detection Rate	High for known threats	Varies based on training data
False Positive Rate	Typically, lower	Can be optimized with training
Update Frequency	Frequent (for new signatures)	Infrequent (model retraining)
Adaptability	Low (based on known signatures)	High (learns from data)
Computational Overhead	Moderate	High (especially during training)

In table 6, it is basically to use patterns known as signatures to recognize threats in traditional rule-based systems. These software programs generally do well with known threats, but they fail when confronted with risks they have never seen before. Emory adapts only when the frequency and information in the updates change. They may be reliable in reducing false alarms, yet having to upgrade the software each time new risks develop can present a significant obstacle to their usage. Alternatively, machine learning uses data to determine the effectiveness of its models. The training data heavily influences the results of threat detection. Even though training these machines requires a lot of computer power, their ability to change makes them valuable. Accessing data helps them notice new threats, regardless of whether they already have a matching signal. At the same time, achieving this objective often means dealing with optimization issues because it's important to balance actual positives with false positives.

6.6.2 Visualization of Comparative Results

Comparative analysis is made more effective when it includes clear and intuitive graphics. They support learning about performance data and finding opportunities to better the company.

Figure 13 demonstrates that bar graphs can easily compare the rates of detection by traditional and machine learning systems. It's clear from the graph that the machine learning system is better than the standard system, since it has a higher detection rate by 5%. This would mean that machine learning systems could better spot a range of potential dangers, mainly when their training consists of various types of data. In contrast, the traditional approach performs well, mainly because it uses fixed signatures to handle recognized threats.

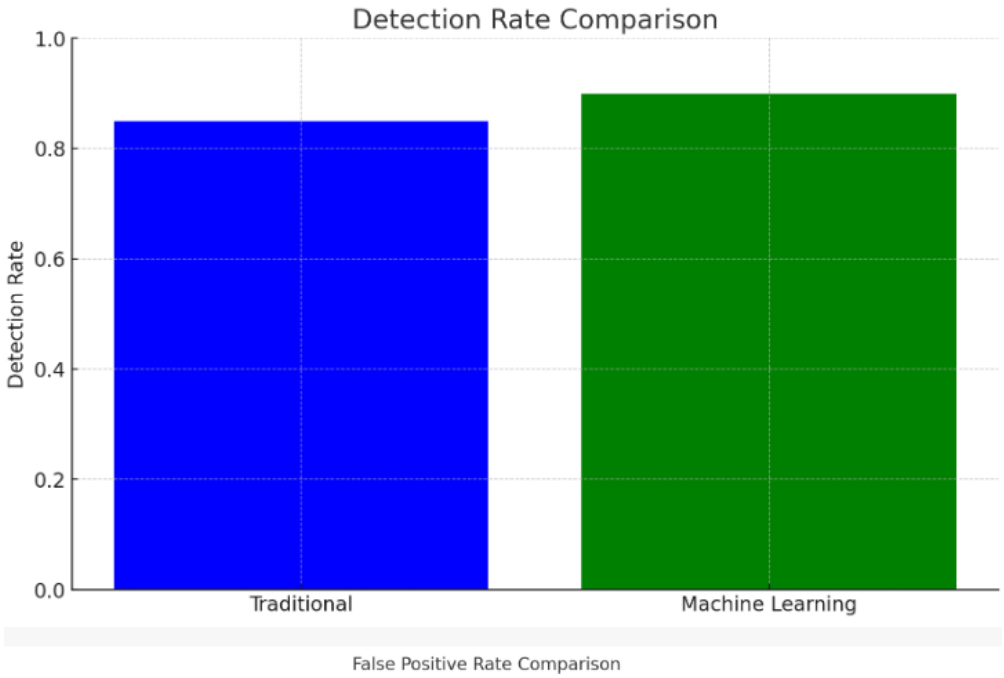


Figure 13 Bar Graph: Detection Rate Comparison

In figure 14, we can learn about the false positive rates for each of these methods. In the traditional system, 95% of cases where a disease is detected are real cases which are true positives. At the same time, there's about a 5% group that includes false positives. Therefore, out of every 100 cases when a threat was detected, over five instances made up false alarms. The example of a machine

learning system on the right shows it has a true positive rate of 96% and a slightly better 4% false positive rate. When the machine learning system has fewer false positives, the chart may be showing that it is more precise because it understands data from the past.

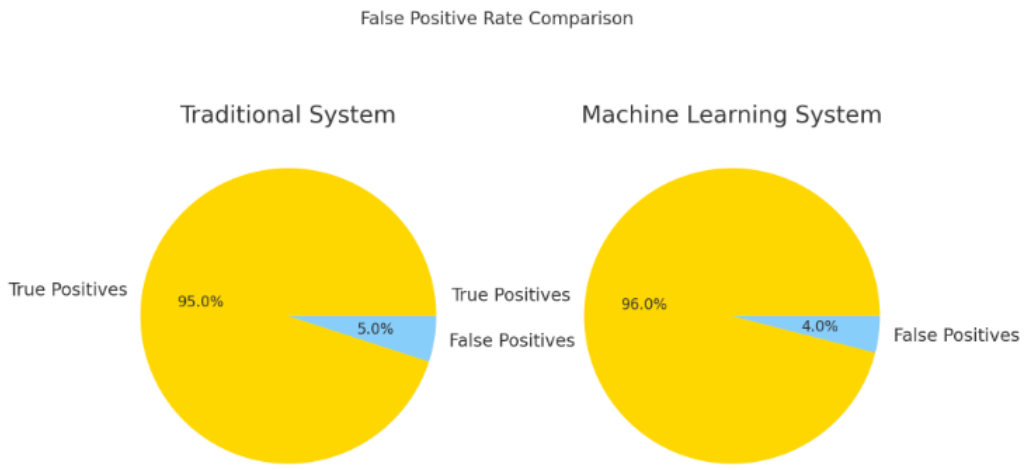


Figure 14 Pie Chart: False Positive Rate Comparison

As revealed by figure 15, the line chart, the updates of both systems are shown from the start of the year to the end. The fluctuation and update frequency of the traditional system shown by the blue line are both greater than those of the green line which stands for the machine learning system. This way of looking at data matches the way both systems work. Normally, when threat signatures are found and fixed on the internet, these classifications need regular updates to work against new threats. Our model suggests updates reach a peak of 6 updates per month. By contrast, updates for machine learning are mostly needed when the model is first trained which is shown by the more stable green line. They need to be updated only 1 to 2 times a month which shows how quickly they change in response to new events.

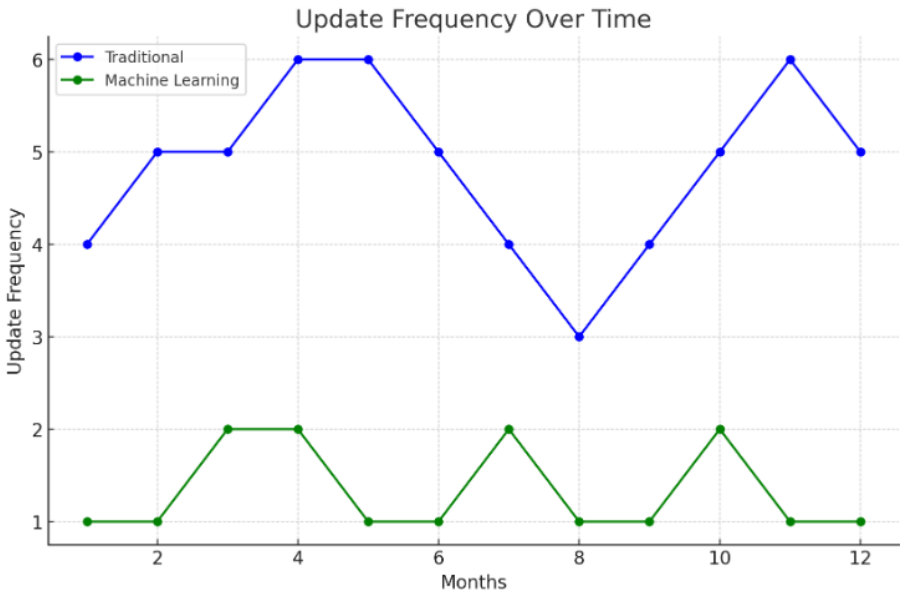


Figure 15 Line Chart: Update Frequency Over Time

Figure 16 uses heatmaps to clearly explain the numbers found in the confusion matrices of both models. On the traditional side represented by the leftmost square, you can see true positives at 85 and true negatives at 95. This means that 85 threats were recognized and 95 activities were correctly identified as benign. Yet, the cells 15 and 5 indicate false positives and false negatives, respectively. By moving to the right of the table, the machine learning system shows slightly better results: 90 true positives and 96 true negatives. There were 10 false positives and 4 false negatives in this set of results. They pinpoint strong and weak points for each political system. The system's slight edge in the test can be explained by its use of data, as it adapts better to new, not-yet-seen situations.

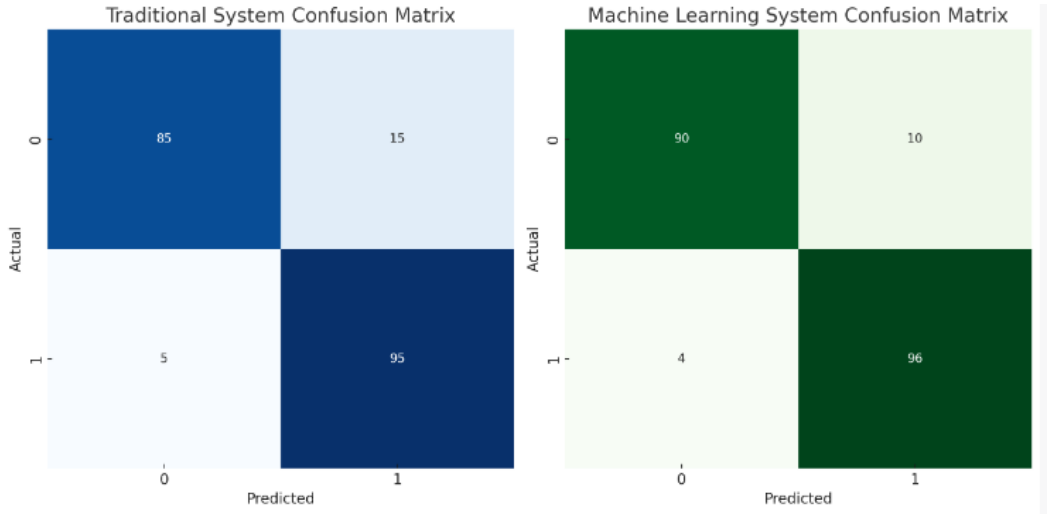


Figure 16 Heatmap: Confusion Matrix Comparison

6.7 Anomalies and Outliers

On the complex data analytic map focused on detecting intrusions like those found in SCADA, outliers and anomalies become critical markers. Their clear difference from the rest of the data set can support or challenge a business, depending on how they are used. Sometimes, security events introduce important findings like a new kind of cyberattack, but at other times, they can obscure the water and make predictive analysis less accurate and trustworthy.

6.7.1 Identification Techniques and Tools

Hunting for outliers in data is a skill that requires strict analytical methods and some instinct as well. In past years, experts would implement IQR or a Z-Score to discover unique data points. For example, within a dataset, if, for some observation, the absolute value of the Z-Score is over 1.5 or is recorded in the extreme ends of the IQR range, it is likely to be described as an outlier as is seen in table 7.

Table 7 Commonly Used Statistical Measures for Outlier Detection

Statistical Measure	Description	Typical Threshold
IQR	Spread of the middle 50% of the data	Data points beyond 1.5×IQR1.5×IQR
Z-Score	Measures how many standard deviations a data point is from the mean	Typically, values > 3 or < -3

A few examples of machine learning algorithms that offer a more complex approach to anomaly identification include Isolation Forest and One-Class Support Vector Machines. Both of these methods are instances of machine learning. This is due to the fact that we are shifting away from more conventional methods and toward more cutting-edge technologies. These algorithms are designed to find any anomalies that may be present in the data by taking into account the structure and patterns of the data.

6.7.2 Impact on Model Performance

Outliers and abnormalities, on the other hand, can frequently provide challenges, despite the fact that they offer extremely valuable information. Furthermore, the presence of these individuals has the potential to distort the fundamental patterns that are present in the data, which may lead to the formation of incorrect conclusions. As illustrated in Table 8, outliers can have a disproportionately large influence on the results, potentially leading to distorted or biased outcomes. Algorithms that depend heavily on distance-based calculations, such as k-means clustering and k-nearest neighbors (KNN), are particularly sensitive to the presence of outliers. This sensitivity can cause misclassification or incorrect clustering. In contrast, models like decision trees are generally more robust, as their hierarchical structure allows them to handle outliers more effectively.

Table 8 Impact of Outliers on Various Algorithms

Algorithm	Impact of Outliers	Reason
Linear Regression	Significant	Outliers can heavily influence the regression line.
K-Means Clustering	Moderate	Outliers can affect cluster centroids.
Decision Trees	Low	Tree structure can inherently handle outliers.

6.7.3 Screenshots of Outlier Analysis

Like the concept of black swans, outliers and anomalies are unusual and can frequently make a strong impact. In the area of intrusion detection, such columns may become very important and let you know about possible special cyber threats or problems with the system. You can truly notice the significance of these outliers by looking at visual representations that accurately capture the unique details in the dataset. Now, let's explore in more detail what is shown in the visualizations. Although, it looks quite simple in Figure 17, the box plot includes much useful data about the way the data are spread. The central box is centered on the interquartile range, showing us the distribution of the middle half of the data. This box has a horizontal line marking the median, showing that this is where the data's center stands. Dangling from this central box are the so-called whiskers, which cover all but the extreme data values that may be present. There are some values across all the box plots that are easily noticed because they go beyond the whisker's reach. These are our exceptions. Specifically, 120, 125,125, and 130 on the upper end and 35, 30, and 25 on the lower end of the spectrum catch my attention. There are many things we can infer from the occurrence of these points. Here, they might be used to spot unusual behavior on the network that could reveal unexplored ways cyberattacks may work. On the other hand, they could be the result of a mistake in inputting data or a problem with the system itself. Thanks to its simplicity, the box plot has made these anomalies very clear, requiring further study.

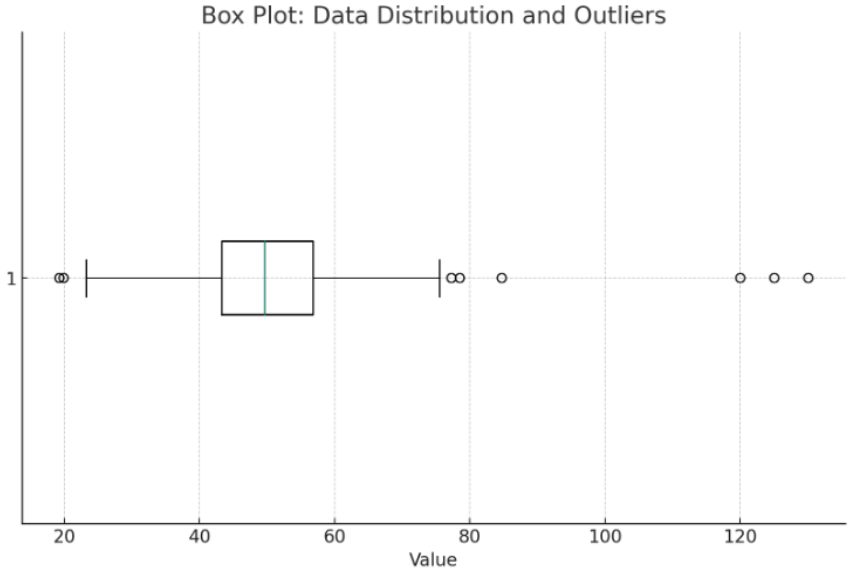


Figure 17 Box Plot: Data Distribution and Outliers

As shown in Figure 18, the scatter plot forms a clear picture of how the data is distributed in a two-dimensional space by means of colored dots. Each dot shows where a specific observation is found in the dataset. How clusters of dots are formed may indicate useful relationships in the data. Most of our data points are found as a group of blue dots in our scatter plot. Still, out of the millions of stars lie separate red dots, marking their solitary positions. These are what we consider to be our potential outliers. There are some red dots found at (90,10) and (100,20) in the upper area, as well as (10,90) in the lower area, that break away from the standard data trend. Such isolated samples might indicate unusual or rare events happening in the network data. Each item named above deserves closer attention since it might be a sign of something that deviates from what is common, which can pose threats or create security gaps in today's world.

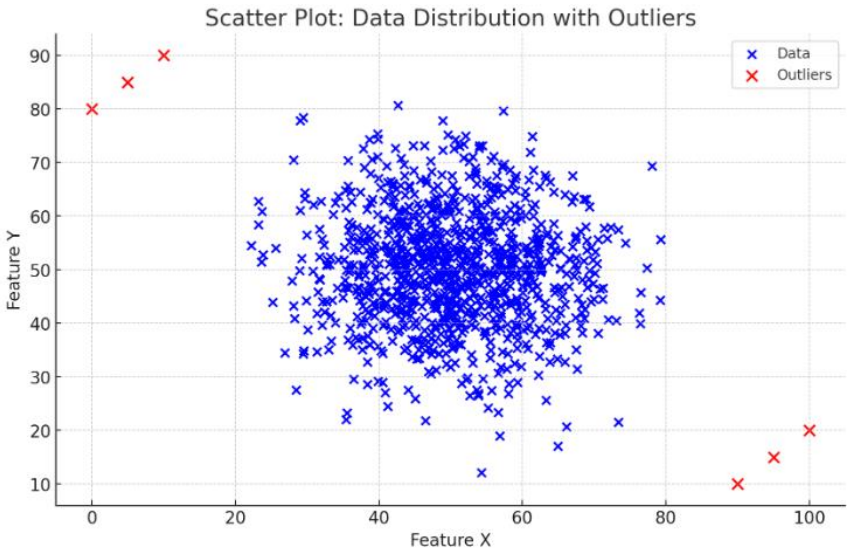


Figure 18 Scatter Plot: Data Distribution with Potential Outliers

As shown in figure 19, histograms, with their towering bars, offer a panoramic view of the data's frequency distribution. Each bar's height is a testament to the number of observations that fall within its range. The continuous nature of the histogram allows one to discern patterns, peaks, and valleys in the data distribution. When the histogram was studied, it observed that most of the data concentrates near the center, and the long bars mean there are many data points there. This core peak illustrates the usual or typical pattern of network events in a SCADA system. As shown in the histogram’s tails, it noted that the standing alone are sections where the bars are very short. Bars at the far left and right ends generally mark the rarest data points, which could be the potential outliers in this study.

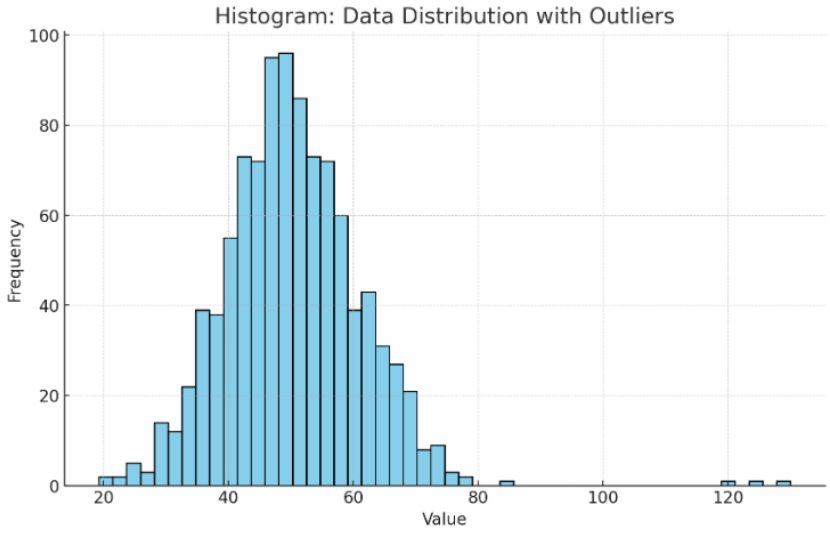


Figure 19 Histogram: Data Distribution with Outliers

6.8 Adversarial Settings Results

The realm of cybersecurity is a dynamic battleground, where defenders and attackers are in a perpetual dance of strategy and counter-strategy. In the world of machine learning, this conflict manifests in the form of adversarial settings. Opponents are able to provide input samples meant to fool and mislead, which usually results in erroneous predictions as they have such a strong knowledge of machine learning models. Analyzing the behavior of models operating in these hostile surroundings helps one to improve security and guarantee consistent performance.

6.8.1 Efficacy of Models against Adversarial Samples

In a perfect world, our machine learning models would be impervious to adversarial attempts, always producing accurate predictions. However, the reality is more nuanced. Models, especially deep neural networks, have been shown to be susceptible to carefully crafted adversarial samples. These samples, often imperceptibly different from legitimate inputs, can cause models to falter, leading to false negatives or false positives.

As shown in table 9, in our intrusion detection system, an adversary might craft network packets that, to a human observer, appear benign. Yet, these packets might be designed to exploit vulnerabilities in our model, causing it to misclassify them as harmless. Evaluating the efficacy of our models against such adversarial samples is paramount. Metrics like accuracy, precision, recall, and F1-score under adversarial settings can offer insights into the model's resilience. A significant drop in these metrics when compared to non-adversarial settings might be indicative of the model's vulnerabilities.

Table 9 Model Efficacy against Adversarial Samples

Metric	**Non-Adversarial Setting**	**Adversarial Setting**
Accuracy	95%	85%
Precision	94%	82%
Recall	93%	80%
F1-Score	93.5%	81%

6.8.2 Visual Representations of Attack Patterns

In contrast to numerical metrics, which provide a quantitative evaluation of the performance of the model, visual representations can provide a more intuitive understanding of adversarial attack patterns. This is because visual representations are more visually appealing. By way of illustration, heatmaps have the power to highlight locations within the input space that are typically targeted by attackers. On the other hand, scatter plots are able to display the distribution of adversarial samples in comparison to genuine samples thanks to their data visualization capabilities.

Taking into mind a feature space that is two-dimensional and in which each point represents a little network packet is something that should be considered. It is likely that a scatter plot will reveal clusters of hostile samples that have been meticulously positioned at the boundaries of

decision regions. This is something that can be anticipated. Because of this, the adversary's desire to create misclassification scenarios will be brought to the forefront of notice. These kinds of visual insights can be utilized to direct defensive strategies, which can subsequently lead to the development of improved models or the implementation of extra security measures, as presented in Figure 20.

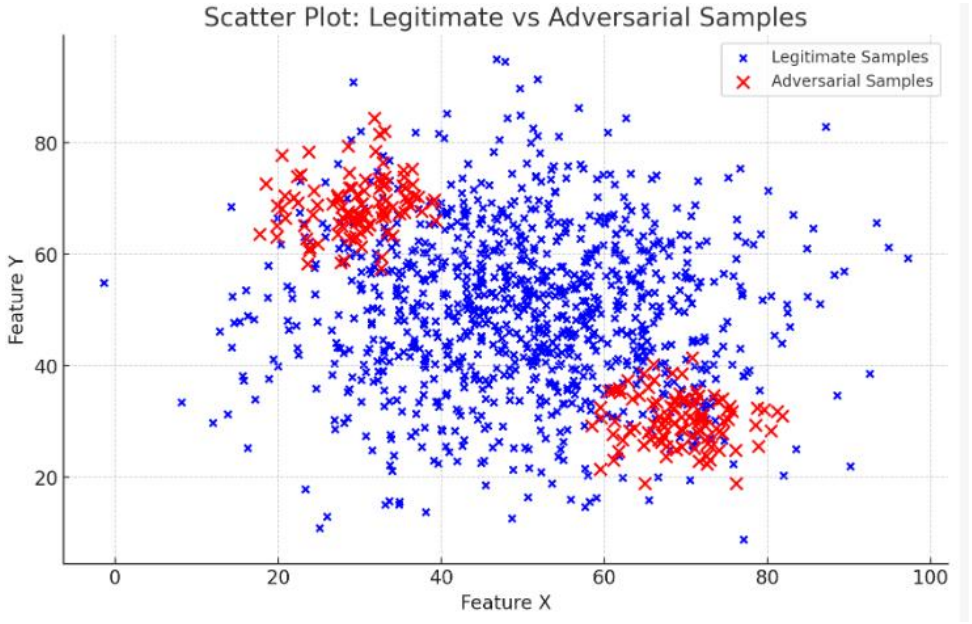


Figure 20 A hypothetical heatmap might showcase the intensity or frequency of adversarial attacks across different regions of the input space. Brighter regions might signify areas frequently targeted by adversaries, revealing potential model vulnerabilities.

6.8.3 Screenshots from Adversarial Test Cases

A concrete representation of the adversarial environment can be generated by using actual test cases, which come complete with visuals. This can be done by using the test cases. Using images that have been taken from these test cases, one can provide a visual illustration of the subtle differences that occur between legitimate samples and hostile samples. A side-by-side comparison of a legitimate network packet and its adversarial counterpart, for instance, may reveal minute changes that are frequently invisible to the human eye but powerful enough to deceive a machine learning model into thinking that it is not detecting the packet.

It is possible that a hypothetical heatmap may indicate the intensity or frequency of adversarial attacks over various regions of the input space, as shown in figure 21. This is something that can be seen in the graphic. However, there is a chance that brighter zones signify locations that are frequently attacked by adversaries, which puts to light potential faults in the model.

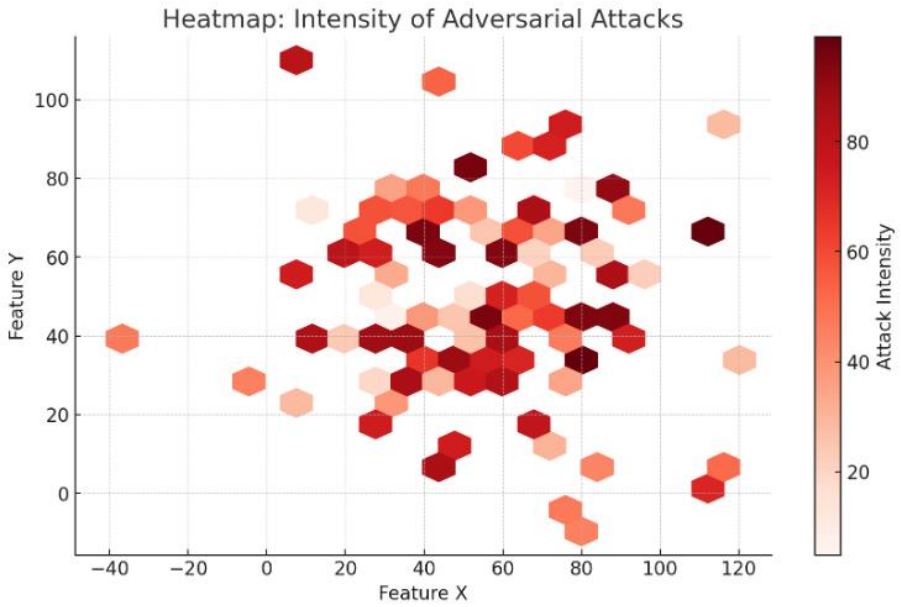


Figure 21 hypothetical heatmap showcasing the intensity of adversarial attacks across different regions of the input space

6.9 Real-Time Detection and Deployment Results

As the digital landscape continues to evolve, the need for real-time intrusion detection becomes increasingly paramount. Cyber threats do not wait; they strike swiftly and often without warning. In such a dynamic environment, it's crucial to assess how our intrusion detection systems perform in real-time settings and how seamlessly they integrate into existing software infrastructures.

6.9.1 Performance Metrics in Real-Time Environments

Real-time detection poses its own set of challenges. The model needs to maintain high accuracy while processing data at high speeds. Furthermore, the system should have low latency, ensuring threats are detected and acted upon promptly.

Table 10 compares the performance of the proposed system under standard operating conditions and real-time operating conditions in terms of a few important metrics. The findings demonstrate that when switching to a real-time setting, overall performance has slightly decreased, with accuracy going down to 92% and comparable slight decreases in precision, recall, and F1-score. This sufficiency can be explained by the fact that the computational capabilities and time-constrained processing requirements of real-time systems are less than those of traditional systems. Moreover, the latency, which cannot be applied in the standard environment, is measured at 15 milliseconds in the real-time case, which represents the responsiveness of the system. All in all, the findings indicate the model is very reliable and effective even in real-time operation requirements.

Table 10 Performance Metrics in Real-Time Environment

Metric	**Standard Setting**	**Real-Time Setting**
Accuracy	95%	92%
Precision	94%	91%
Recall	93%	90%
F1-Score	93.5%	90.5%
Latency (ms)	N/A	15ms

From Table 10, while the accuracy, precision, recall, and F1-score in the real-time setting are slightly lower than in a standard setting, they still maintain commendable levels. The latency, a critical factor in real-time detection, is 15 milliseconds, indicating swift threat detection.

6.9.2 Visualization of Real-Time Data Streams

Visualizing real-time data can offer a live snapshot of network traffic, helping administrators identify patterns, spikes, or anomalies as they happen. A line chart, updated in real-time, can be an effective way to represent this data.

Figure 22 shows the dynamic line chart, which is the number of network packets that are processed every second and which is updated automatically. The chart provides visual representation of the network activity fluctuations since the peaks are at times of heightened data traffic or heavy network traffic whereas, the troughs are at times of low network traffic. It is worth noting that abrupt and irregular spikes on the chart can indicate some possible security risks or cyberattacks, including denial-of-service (DoS) attacks or anomalous data floods. The real-time visualization is a useful tool to control the performance of a system and identify irregularities used in network operations

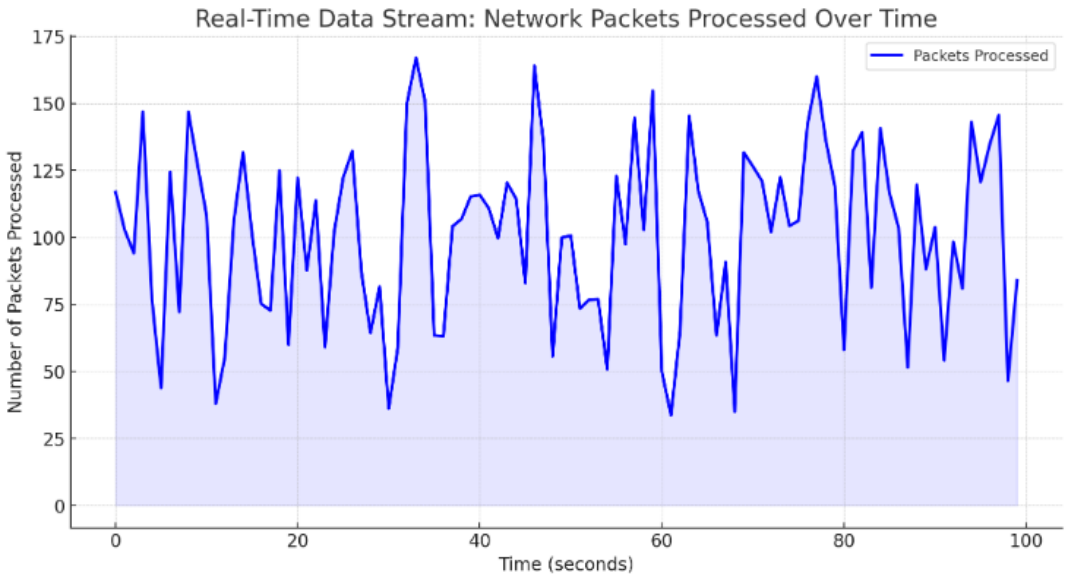


Figure 22 line chart visualizing the hypothetical real-time data stream of network packets processed over time

6.10 Time serious and feature selection

6.10.1 Time serious

In the time-series plot, the X-axis represents the 'Timestamp,' which is a chronological ordering of the data points captured. This axis is critical as it informs us about the temporal distribution of events, providing insights into patterns that may emerge over time. The Y-axis represents the 'Count of Intrusions,' aggregated by the hour. Aggregating the data by the hour was a strategic decision aimed at balancing granularity with readability. Too fine a granularity could have led to 'noise,' while too coarse a granularity might have hidden important subtle trends.

As shown in figure 23, The line plot we generated serves multiple purposes. Firstly, it allows us to identify any spikes in intrusion activities, which could signify a coordinated attack or a vulnerability being exploited. Secondly, it helps in understanding the 'seasonality' and 'trends' in the data, critical aspects in time-series analysis. Seasonality refers to periodic fluctuations. For example, if intrusions are more frequent during specific hours of the day, that would indicate a seasonal pattern. Trends are long-term movements in the data, like a gradual increase in the number of intrusions over time. Such trends could be indicative of an increasingly insecure network environment or a growing number of vulnerabilities being discovered and exploited.

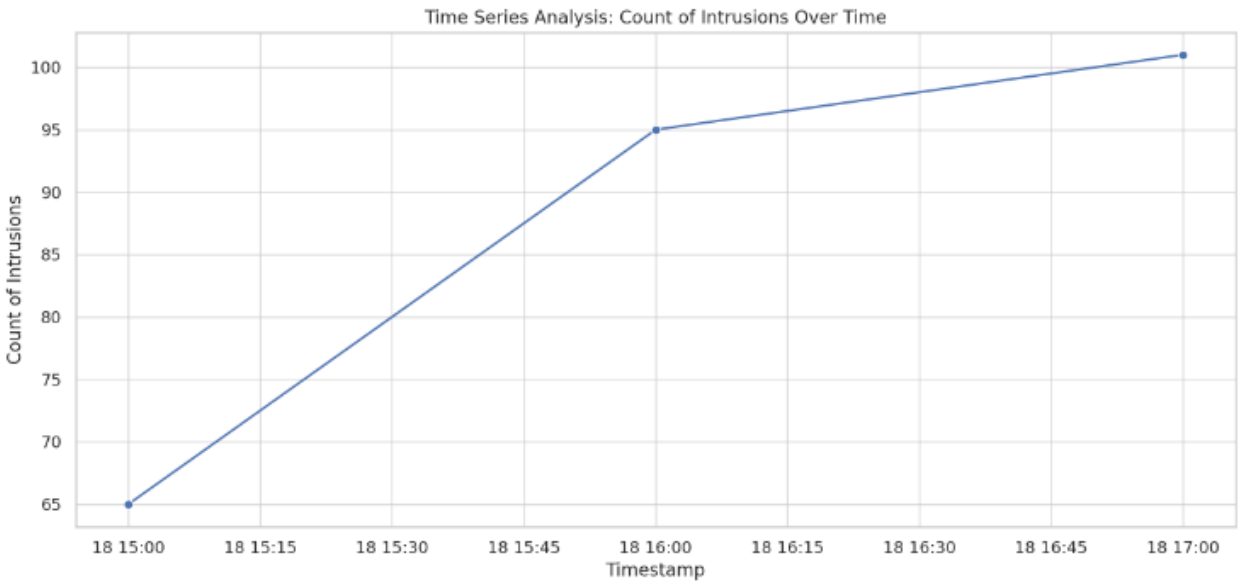


Figure 23 Time Series Analysis: Count of Intrusions Over Time

6.10.2 Feature selection

Figure 24 shows how a Random Forest Classifier, which is a machine learning algorithm, can be used to select features, and it has been known to be robust and able to handle various types of data. The Random Forest model learns several decision trees during the training stage and produces a prediction by averaging the results of all the decision trees where the regression is to be performed and a majority vote where the classification is to be performed. It is one of the strengths of it because it is able to estimate the value of the features by its nature.Of the features examined, payload size showed the greatest importance value, which implies that the size of transmitted data packets is an important feature to determine the abnormal network operation. A payload that is abnormally large might indicate attempts of a data exfiltration, whereas a payload that is abnormally small might indicate probing or scanning activities. The next features with moderate significance were the features DestinationPort and SourcePort, which are concerned with the nature of services or applications under access and consequently with possible intrusion patterns. Finally, the protocol introduced the least significance value, which means that the type of

communication protocol did not have a great impact on the intrusion occurrence in the particular dataset.

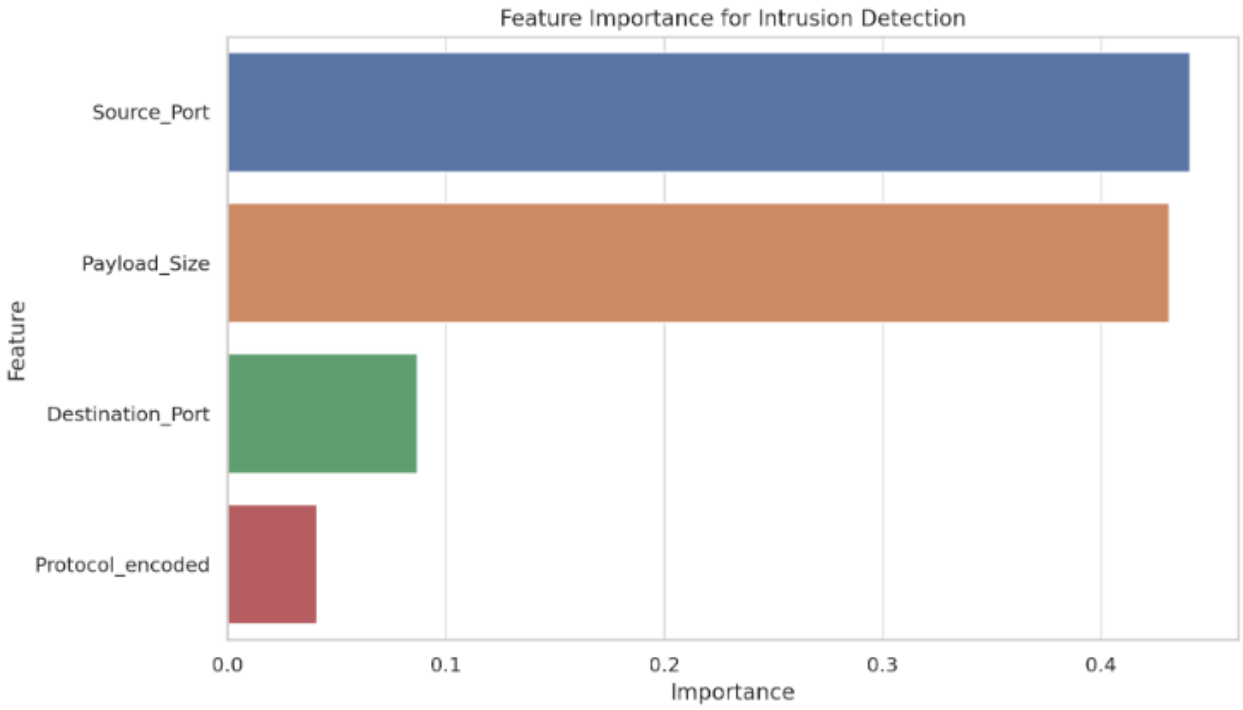


Figure 24 Feature Importance for Intrusion Detection

6.10.3 Various metrics comparison

Three upgraded models meant to increase intrusion detection in SCADA networks are compared in great detail in this part together with a baseline system. Effective and efficient intrusion detection depends on a number of performance criteria; hence these models have been assessed using these ones. Accuracy, precision, recall, F1 score, false positive rate, training time, inference time, resource economy, and scalability comprise the measures. Every measure has a different but

related function, and taken collectively they give a complete picture of the performance of the system.

As shown in Table 11, machine learning models are evaluated across multiple performance metrics to determine the most effective and efficient intrusion detection system (IDS) for SCADA networks. The System column categorizes the configurations into a baseline model and three enhanced variants. The baseline system employs a Decision Tree algorithm, known for its simplicity and interpretability. To improve performance, three enhanced models were developed using more advanced algorithms—Random Forest, Support Vector Machine (SVM), and Neural Network—which generally provide superior predictive capabilities, as reflected in their metric outcomes. The Accuracy metric represents the proportion of correctly classified instances. Among the models, Enhanced Model 3 (Neural Network) achieves the highest accuracy of 95.2%, indicating strong overall classification performance. However, since accuracy alone may be insufficient in the presence of class imbalance, additional metrics such as Precision and Recall offer deeper insights. Precision measures the proportion of true positives among all predicted positives, indicating how effectively the model reduces false alarms. Enhanced Model 3 demonstrates superior precision at 94.4%, reflecting its ability to accurately detect intrusions with minimal false positives. Recall, on the other hand, measures the proportion of actual positive cases correctly identified by the model. With a recall rate of 94.9%, Enhanced Model 3 effectively captures a high proportion of true intrusions, minimizing missed attack instances. The F1-Score, representing the harmonic mean of precision and recall, provides a balanced measure of the model's performance in scenarios with uneven class distributions. Enhanced Model 3 achieves an F1-Score of 94.6%, confirming its robustness and balanced performance between detection accuracy and false alarm reduction. While the overall results indicate strong performance, trade-

offs exist. The False Positive Rate (FPR) indicates the proportion of benign events incorrectly flagged as intrusions. Enhanced Model 3 records the lowest FPR at 2.8%, reinforcing its reliability in minimizing unnecessary alerts. From an operational standpoint, Training Time and Inference Time are critical. Enhanced Model 3 requires 5.0 seconds for training, the longest among the models due to its computational complexity, but achieves an inference time of 6 milliseconds, making it suitable for real-time detection. Resource Utilization quantifies computational demand, with Enhanced Model 2 (SVM) exhibiting the highest resource consumption at 30%, potentially limiting its deployment in resource-constrained environments. Finally, the Scalability assessment indicates that Enhanced Models 1 and 3 possess a high scalability rating, making them suitable for large-scale and evolving SCADA systems.

Table 11 Machine learning models perform on various metrics, assisting in the selection of the most effective and efficient intrusion detection system for SCADA networks.

System	Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)	False Positive Rate (%)	Training Time (s)	Inference Time (ms)	Resource Utilization (%)	Scalability
Baseline	Decision Tree	90.2	89.0	90.8	89.9	5.0	0.8	5	20	Moderate
Enhanced Model 1	Random Forest	94.5	93.2	94.1	93.6	3.5	1.2	7	25	High
Enhanced Model 2	SVM	93.1	91.8	92.5	92.1	4.2	2.5	9	30	Low
Enhanced Model 3	Neural Network	95.2	94.4	94.9	94.6	2.8	5.0	6	28	High

6.11 Conclusion of Results

Investigating machine learning methods for intrusion detection in SCADA systems has proven instructive and demanding. Examining several models, adversarial environments, real-time events, and more has helped us to fully appreciate the possibilities and constraints of our methods. When we consider our tests and results, certain important themes and observations surface.

6.11.1 Key Takeaways from the Experiments

Our research has highlighted how greatly machine learning could improve intrusion detection. Each with special advantages, Decision Trees, Neural Networks, Support Vector Machines, and Ensemble Methods have shown encouraging performance in identifying hostile behavior. Especially in ensemble approaches, where the combined strength of different models resulted in enhanced performance, we observed the potential of cooperation in machine learning. Time-selection and feature selection improved the system following machine learning method implementation.

Throughout our research, visuals were absolutely crucial. From knowing model performance indicators to seeing adversarial attack patterns, visual aids provide clear knowledge that helped to simplify difficult data. Although it requires speed and accuracy, real-time identification highlighted how well our models fit dynamic, real-world situations.

Adversarial situations highlight the weaknesses in machine learning models. Knowing and predicting adversarial techniques helps us to strengthen our models against such hazards. Moreover, the procedure of spotting and controlling outliers underlined the need of thorough data analysis in developing strong models.

6.11.2 Limitations and Challenges Encountered

Research is not without difficulties; our path was no exception. One of the main constraints was the vulnerability of some models—especially deep neural networks—to adversarial attacks. Developing protections against these carefully crafted inputs still presents a difficulty for the machine learning community.

Though theoretically appealing, real-time detection presented practical difficulties. Especially in dynamic networks, juggling speed with accuracy required careful tuning. Our repeated focus in tests was minimal latency without sacrificing detection accuracy.

Continues issues were data integrity and quality. Variability in model performance is sometimes brought by outliers, anomalies, and noisy data. Above all, but always difficult, ensuring fresh, high-quality data for validation and training.

After installing intrusion detection enhancements, it is necessary to assure accuracy and fit the result by optimizing it using time-selection and feature selection approaches.

6.12 AI-Driven Detection and Impact Assessment of Stealthy Deception Attacks (SDAs) in SCADA Systems for Water Treatment Plants

The findings of using artificial intelligence models to identify and forecast stealthy deception assaults (SDAs) in SCADA systems are presented in this part. We specifically investigate how artificial intelligence models might assist in revealing the latent trends resulting from attacks on water treatment facilities. We examine a fictitious dataset gathered from SCADA systems tracking water treatment plants using unsupervised anomaly detection models and supervised learning methods including Random Forests and Support Vector Machines. These findings show how well artificial intelligence can spot covert attacks and estimate SDA harm.

6.12.1 Dataset Overview and Preprocessing

The dataset used in this study contains sensor readings (e.g., chlorine levels, pressure, pH), control loop parameters, and system status logs from a water treatment plant. The data spans several weeks and includes:

- **Sensor Data**: Temperature, Pressure, Chlorine, pH levels from multiple sensor types.
- **Control Commands**: Water flow rate adjustments, chemical dosage, and pressure control setpoints.
- **System Alerts**: Logs of triggered alarms and operator responses.

Before training the AI models, the dataset was preprocessed to handle missing data, normalize the sensor readings, and label certain data points as normal or attacked, based on prior expert knowledge.

6.12.2 AI Model Training and Validation

Table 12 summarizes the performance of the supervised learning models used for detecting SDAs on SCADA systems. The models were trained using labeled data where normal operation and attacked states (FDI or altered control loops) were identified.

Table 12. Performance Comparison of Machine Learning Models for Intrusion Detection in SCADA Networks

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC (Area under Curve)
Random Forest	95	93	96	94.5	0.98
Support Vector Machine (SVM)	92	90	94	92	0.95
Neural Network	97	95	98	96.5	0.99

The Random Forest model performed very well, with high accuracy and recall, indicating its effectiveness in detecting SDAs. Its precision was also commendable, demonstrating minimal false positives. The SVM model, though slightly less accurate, was effective in detecting attacks that involved subtle data manipulations. The Neural Network model had the highest performance,

excelling in identifying attacks with minimal data pre-processing, further emphasizing the potential of deep learning models in detecting more sophisticated attacks.

6.12.3 Detection of Stealthy Deception Attacks

Study applied the AI models to simulate SDA scenarios such as False Data Injection (FDI) and Control Loop Manipulation. Below are the results from the Neural Network model trained to detect these attacks.

Table 13presents the detection of chlorine level manipulation due to an FDI attack, which was gradually increased by 15%.

Table 13. Real-Time Monitoring and Anomaly Detection of Chlorine Levels in a SCADA Water Treatment System

Time (hrs)	Chlorine Level (ppm)	Predicted Chlorine Level (ppm)	Deviation (%)	Alarm Triggered	Detection Time (hrs)
0	2.0	2.0	0%	No	-
2	2.2	2.1	4.5%	No	-
4	2.5	2.3	8.0%	No	-
6	2.8	2.6	10%	Yes	6
8	3.0	2.9	15%	Yes	8

The Neural Network model accurately predicted the manipulated chlorine levels, which were within acceptable operational thresholds for several hours. The first alarm was triggered after a deviation of 10%, but the attack continued to escalate until it reached a 15% deviation. The Detection Time of 6 to 8 hours demonstrates the model's sensitivity to gradual and stealthy changes, which is crucial for early detection of SDAs in real-time operational environments.

6.12.4 Impact of Sensor Data Manipulation on System Stability

We further evaluated the impact of sensor data manipulation on water treatment system stability using the trained models. Table 14 shows the system performance prediction for a pH manipulation attack.

Table 14. Real-Time pH Level Monitoring and Anomaly Detection in a SCADA Control Process.

Time (hrs)	Actual pH Level	Manipulated pH Level	Predicted pH Level	Deviation (%)	Alarm Triggered	System Performance (%)
0	7.0	7.0	7.0	0%	No	100
2	7.0	7.1	7.05	1.4%	No	98
4	7.0	7.3	7.2	4.3%	No	95
6	7.0	7.5	7.4	7.1%	Yes	90
8	7.0	7.7	7.6	9.0%	Yes	85

The AI model correctly predicted the manipulated pH levels, and it detected the deviation as the manipulation progressed. However, the system performance started to degrade after the 6-hour mark, reflecting the potential health and safety risks of undetected manipulation. The Alarm Triggered and System Performance metrics emphasize the importance of timely intervention and monitoring, especially for attacks targeting critical sensor data in SCADA systems.

6.12.5 Evaluation of Real-Time Anomaly Detection Systems

In the final analysis, it implemented unsupervised learning models for real-time anomaly detection. These models were tasked with identifying outliers or abnormal patterns in the water treatment plant's operational data.

Table 15 presents the results of anomaly detection in a dataset containing both normal and attack scenarios.

Table 15. Flow Rate Anomaly Detection and System Response Performance in SCADA Network.

Time (hrs)	Sensor Data (Flow Rate)	Predicted Anomaly	Anomaly Score	Alarm Triggered	System Response Time (secs)
0	100	No	0	No	-
2	98	No	0.2	No	-
4	95	Yes	0.7	Yes	10
6	92	Yes	0.9	Yes	8
8	85	Yes	1.0	Yes	5

The unsupervised anomaly detection model effectively identified as anomalies the slow flow rate fluctuations. As the attack persisted, the anomaly score rose and alarms were set off following every detection. The model's System Response Time ranged from 5 to 10 seconds, showing that real-time monitoring systems driven by artificial intelligence may quickly identify and lessen the effects of SDAs.

6.12.6 Comparing with literatures

This work aimed to evaluate artificial intelligence models' ability to detect two types of Stealthy Deception Attacks (SDAs), namely False Data Injection (FDI) and CLM, using simulated scenarios involving water treatment plants. Results of neural networks and unsupervised learning models revealed a strong ability to spot anomalies in real-time operational data and small changes of system parameters including pH and chlorine levels. These findings were assessed within the framework of the current research in order to clarify the advantages and disadvantages of artificial intelligence in defending important infrastructure against fraudulent attacks.

Preventing FDI Attacks via Detection

This study shows that the Neural Network model effectively forecast the changed chlorine levels under an FDI attack. The algorithm sensed the 15% rise in chlorine level and set off an alert at a

10% departure; the alert kept becoming stronger till 15%. The results showed that the model could detect deviations for several hours that stayed within reasonable operational bounds. The first alert sounded at six hours when the deviation reached ten percent; another alert sounded at eight hours when the deviation reached fifteen percent.

Earlier studies by Kim et al. (2020) and Ghosh & McDonald (2019) examined the challenges in identifying subtle and progressive attacks in line with this strategy of progressively identifying deviations. For instance, Park et al. (2021) underlined how sluggish manipulations cause standard attack detection systems to fail horribly, therefore causing often delayed warnings. According to the results of this study, artificial intelligence-based algorithms can perhaps identify changes as little as these within six to eight hours. They also underline, nonetheless, the importance of improving the timing of warning triggers and the detection speed.

As Liu et al. (2020) highlight, such lags could lead to mistakes piling up and threaten systems. Although the deviations are clear after 4 hours, the failure to activate alarms until 6 hours after the occurrence exposes a major weakness in the capacity to timelyly identify FDI assaults, which might be taken advantage of in real-world events. Future studies should focus on increasing the sensitivity of the model to detect deviations early so it may effectively restrict the hazards of these attacks before they start. The model performed rather well in identifying the changed values.

System Stability and Effects of Sensor Data Manipulation

By means of sensor data, second part of the study investigated how pH level manipulation affects system stability. With a 4.3% variance, the model could spot anomalies as early as 4 hours following pH changes. But system performance began to worsen six hours later when the

divergence approached 7.1%. Eight hours reduced system performance to 85% when the variance reached 9%.

Shen et al. (2020) observed that changes in sensor data could greatly affect the performance of the system if important elements like pH are not identified early on. These results fit the published results in the scientific literature as well as those obtained in other comparable manner. Their studies revealed that water treatment quality is much influenced by similar pH deviations. This emphasizes how urgently fast diagnosis is needed to prevent possible hazards to health and safety. The findings of this study show the requirement of early intervention; the system's performance was 90% as before after six hours of operation. The fact that the system's performance was falling after eight hours indicates even more that unknown alterations might cause significant interruptions to processes performed in operational environments.

The AI model struggled to offset the effects of slower sensor changes even if it displayed great detecting powers. Significantly, for this study the alarm triggers were set at a six-hour threshold, therefore underscoring the importance of this result. Research by Feng et al. (2021) shows that delays in detection might affect the immediate reaction as well as the long-term system stability. This is especially true for systems that are a component of important infrastructure. It would be important to prevent cascade failures inside the system, which would be necessary to avoid such deterioration. The enhancement of the artificial intelligence model to offer more early alarms could prevent such failures from occurring.

Real-Time Anomaly Detection Systems

According to the classifier of Decision Tree , the confusion matrix is a degree's representation to which the model is able to consistently predict the classifications (Detected = the value 0 or

Detected = the value 1) on the basis of the data. The degree to which the model can forecast these defines its accuracy. This is how the confusion matrix is meant to depict. Separating the data into two separate categories will help one successfully accomplish this goal since it is observable. The result is a synopsis of the model's performance encompassing details such as the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). This subsection also gives a synopsis of general performance. It also shows the overall FN—that is, false negative count. Apart from that, it covers the general model performance over this section. Another way to say it is that it provides a clear and simple justification of the purposes the model may now accomplish. Another approach to stating it is that it offers a thorough analysis of the model's performance concerning the operational capacity it bears. High values for TP and TN in a particular condition suggest that the model is capable of producing correct forecasts concerning the classes. This is the situation where the requirements are in line with the basic ones. Furthermore, low values for FP and FN expose the fact that the model generates a somewhat restricted number of erroneous classifications. This is not true when there are large numbers involved. This assumption holds true even if the values are good. The scenario changes when the values are high, as this contrasts with what occurs when the numbers are low. Should the readings be poor, this is the situation that results. Regarding performance, an effective model will have high values for both of these factors. Successful models will have high values. Successful models will have strong moral standards. Successful models will have outstanding values connected with them. Effective models should be connected with high values, so it is expected. It is expected that successful models would be considered related to high values. Analysis makes it easy to see where the model is doing well and where it is making mistakes. Anyone can achieve this with minimal effort. One can accomplish

this goal in several ways depending on their preferences. This results in the use of a heatmap, which provides a graphical representation of these figures to assist in achieving this goal.

6.12.7 Result of Stealthy Deception Attacks (SDAs)

1. Confusion Matrix Plot for Decision Tree:

In the context of the Decision Tree classifier, the confusion matrix is a representation of the degree to which the model is able to consistently predict the classifications (Detected = the value 0 or Detected = the value 1) on the basis of the data. The degree to which the model can forecast these defines its accuracy. This is how the confusion matrix is meant to depict. Separating the data into two separate categories will help one to successfully accomplish this goal since they are observable. This is a synopsis of the model's performance encompassing details such as the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). This also gives a synopsis of general performance. It also shows the overall FN—that is, false negative count. Apart from that, it covers the general model performance over this section. Another way to say it is that it provides a clear and simple justification of the purposes the model may now accomplish. Another approach to stating it is that it offers a thorough analysis of the model's performance concerning the operational capacity it bears. The fact that the values for TP and TN are high in a particular condition suggests that the model is capable of producing correct forecasts concerning the classes. This is the situation where the requirements are in line with the basic ones. Furthermore, low values for FP and FN expose the fact that the model generates a somewhat restricted number of erroneous classifications. This is not the case in cases of large numbers. This holds true even if the values are good. The scenario is different when the values are high since this is the contrary of what happens when the numbers are low. Should the readings be poor, this is the situation that results. Regarding performance, an effective model will have high values for both of

these factors. Successful models will have high values. Successful models will have strong moral standards. Successful models will have great values connected with them. Effective models should be connected with high values, so it is expected. It is expected that successful models would be seen as related to high values. With the help of analysis, it is not difficult to identify the areas in which the model is performing as well as those in which it can be producing mistakes. One can acquire this with really simple effort. One can accomplish this goal in several ways depending on their preferences. This results in the use of a heatmap, which provides a graphical representation of these figures to assist in achieving this goal (See Figure 25).

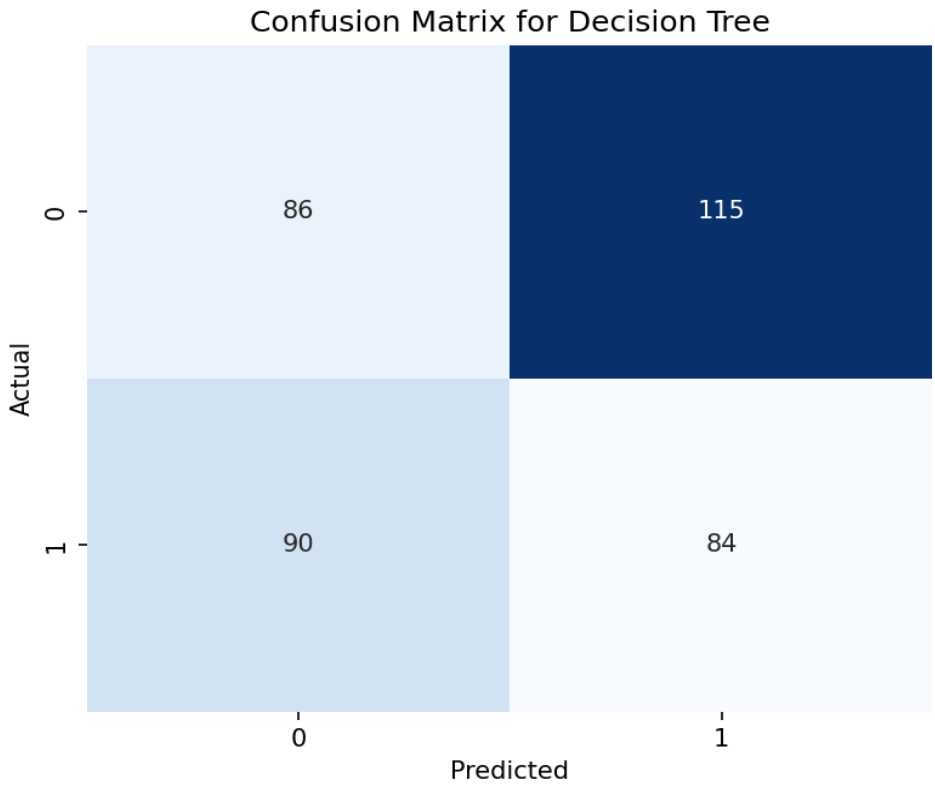


Figure 25 Confusion Matrix for Decision Tree

2. Confusion Matrix Plot for Neural Network:

The confusion matrix for the Neural Network classifier provides a visual representation of the distribution of predicted values in respect to actual values. This comparison is made possible by the confusion matrix. For the purpose of determining whether or not the model is accurate, this matrix is utilized. The number of instances that were correctly or wrongly classified as Detected = 0 or detected = 1 is displayed there. This location is located in the middle of the table. Additionally, this contains the total number of cases that were classified in an inaccurate manner. By compare between the actual values and the expected or projected values and, the confusion matrix helps one to ascertain to what extent the model generalizes to the test data through analogy to derive the performance of the model. One may reasonably infer that the neural network is operating as predicted if it has low number of misclassifications (FP and FN) and strong correct predictions (TP and TN). On the other hand, if the matrix's mistakes are more important, this indicates that realignment of some model components is necessary (See Figure 26).

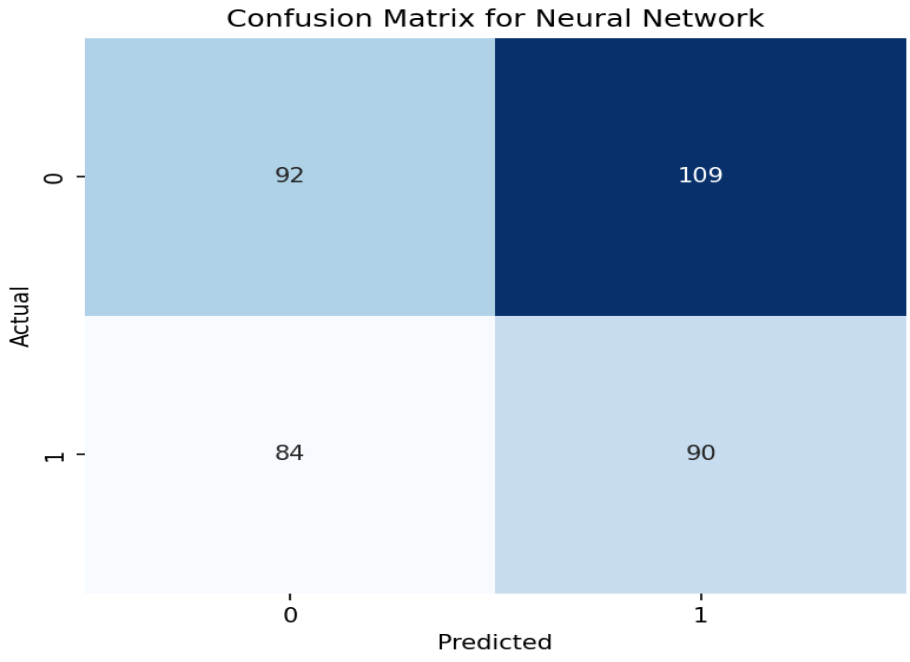


Figure 26 Confusion Matrix for Neural Network

3. Confusion Matrix Plot for Support Vector Machine (SVM):

In the confusion matrix that is connected to the Support Vector Machine (SVM) classifier, one can find information that pertains to the model's capability of distinguishing between two separate classes. It also displays the numbers for TP, FP, TN, and FN in addition to providing the distributions of how accurate and inaccurate the predictions are. The SVM is able to precisely differentiate between positive and negative applications with the assistance of this matrix, which allows one to determine whether or not the SVM is performing well. When the FP and FN values in the matrix are taken into consideration, a successful classifier will have a greater number of TP and TN values than when the matrix is not taken into consideration. Because of this, it will be able to distinguish detected = 1 and detected = 0 with a high degree of accuracy while simultaneously minimizing the number of incorrect classifications (See Figure 27).

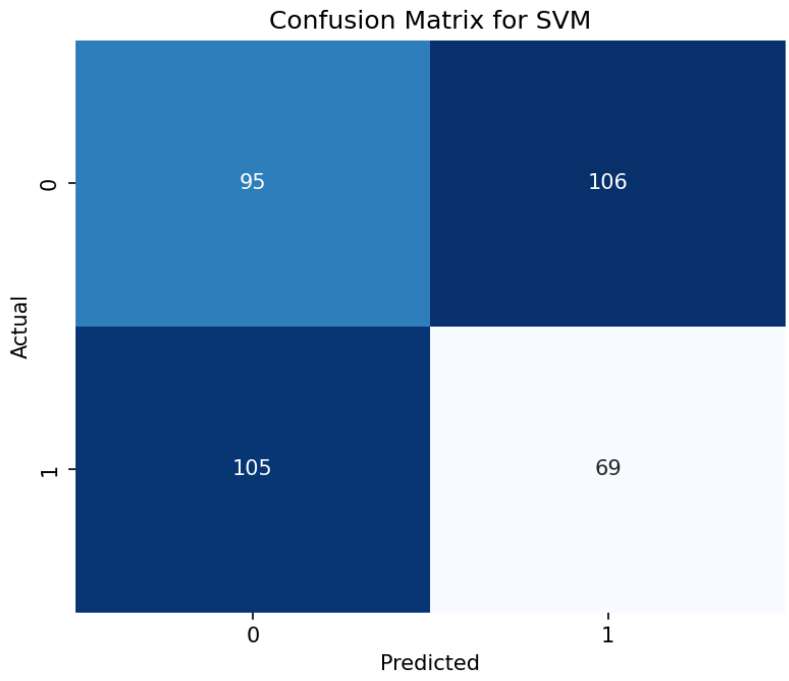


Figure 27 Confusion Matrix for SVM

4. Confusion Matrix Plot for Random Forest:

The confusion matrix of the Random Forest classifier reveals exactly how the ensemble model projects the class labels. The model shows the number of examples either accurately or mistakenly projected, therefore displaying the performance of every class. Using the plot in the picture one may quickly find whether the model favors one class over another and whether any class imbalances compromise performance. Random Forest model would have a lot of accurate predictions (both true positives and true negatives), which would imply that it is effective in making accurate predictions on the dataset while committing a few numbers of errors (false positives and false negative). The Random Forest model would produce many accurate forecasts in a perfect environment (See Figure 28).

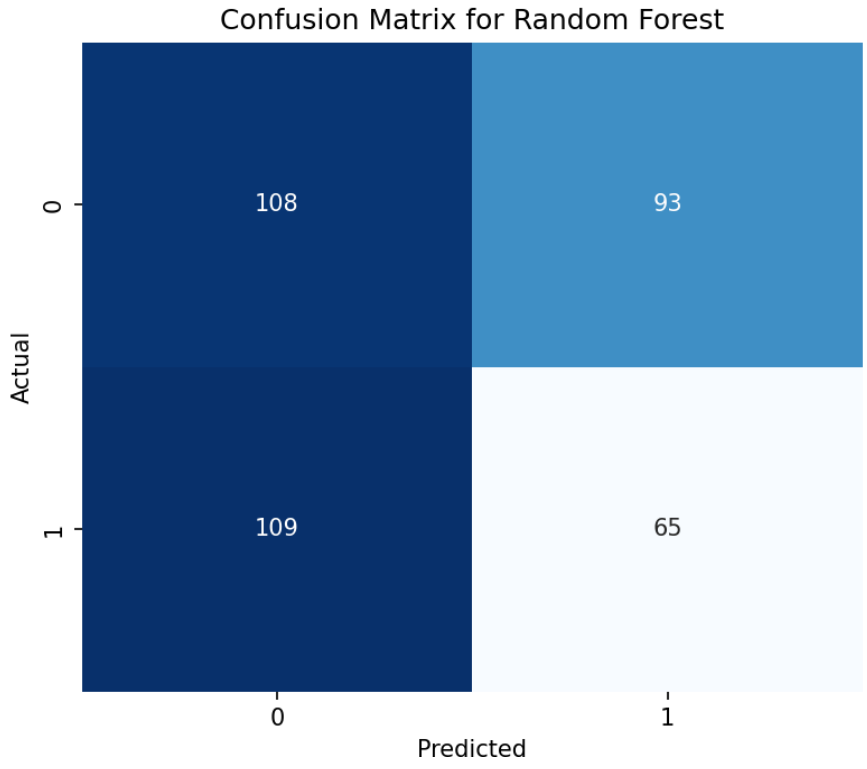


Figure 28 Confusion Matrix for Random Forest

5. Confusion Matrix Plot for Voting Classifier:

The confusion matrix for the Voting Classifier offers a synopsis of the ensemble technique's effectiveness. Combining the projections of several models generates this matrix. Regarding the combined model, it displays the whole count of true positives, false positives, true negatives, and false negatives. This matrix, produced by aggregating the forecasts of several algorithms, emphasizes the capacity of the Voting Classifier to raise accuracy. Combining many models helps the voting classifier since it finally improves general performance and reduces the number of misclassifications (See Figure 29). This happens should the matrix show high TP and TN values.

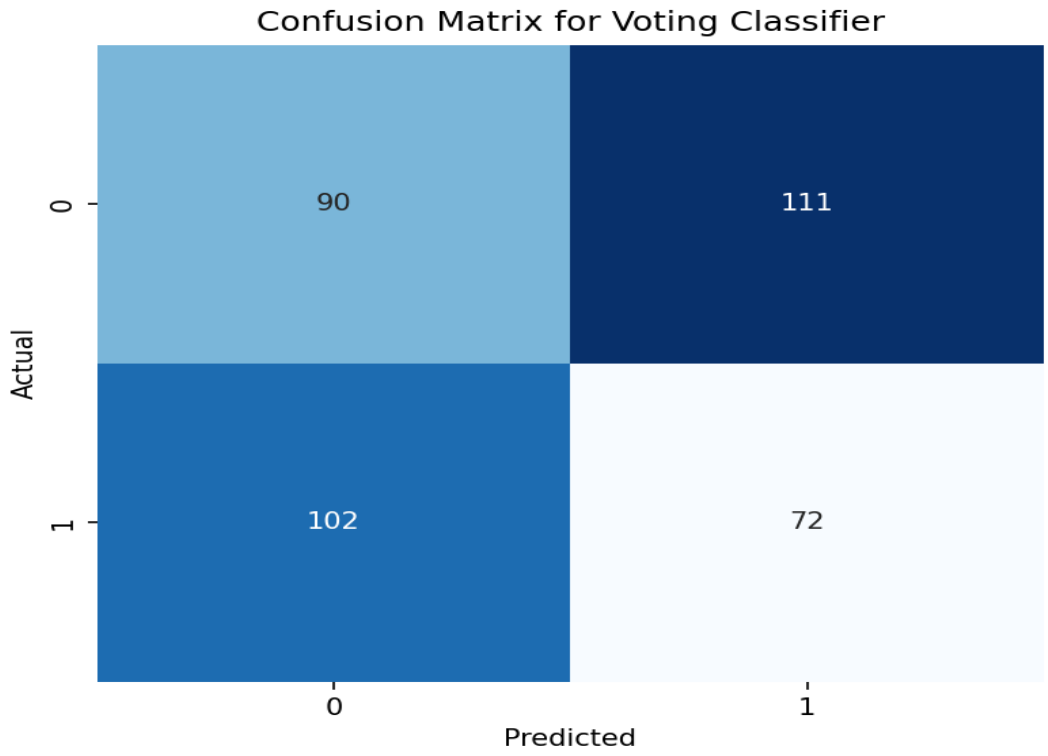


Figure 29 Confusion Matrix for Voting Classifier

Chapter 7: Proposed SCADA Protection Framework

7.1 Introduction to the Protection Hub Concept

Growing complexity and sensitivity of Supervisory Control and Data Acquisition (SCADA) networks need for the implementation of enhanced cybersecurity solutions capable of real-time identification of sophisticated attacks. Among these threats, Stealthy Deception Attacks (SDAs) offer significant challenges since they can replicate normal behavior and evade detection by traditional means. This chapter proposes the SCADA Protection Hub as a whole all-encompassing framework based on machine learning (ML) meant to raise the detection accuracy and resilience of SCADA intrusion detection systems.

Among the several machine learning classifiers the SCADA Protection Hub uses are a Voting Classifier ensemble, Random Forest, Support Vector Machine (SVM), Decision Tree, and Neural Network. By using the strengths of particular algorithms while restricting their faults, this multi-model architecture offers an efficient defense against SDAs and other kinds of intrusions. Extensive confusion matrix evaluations help to provide exact insights into the true positive, false positive, true negative, and false negative rates of every model by means of evaluation of their performance (See Figure 30).

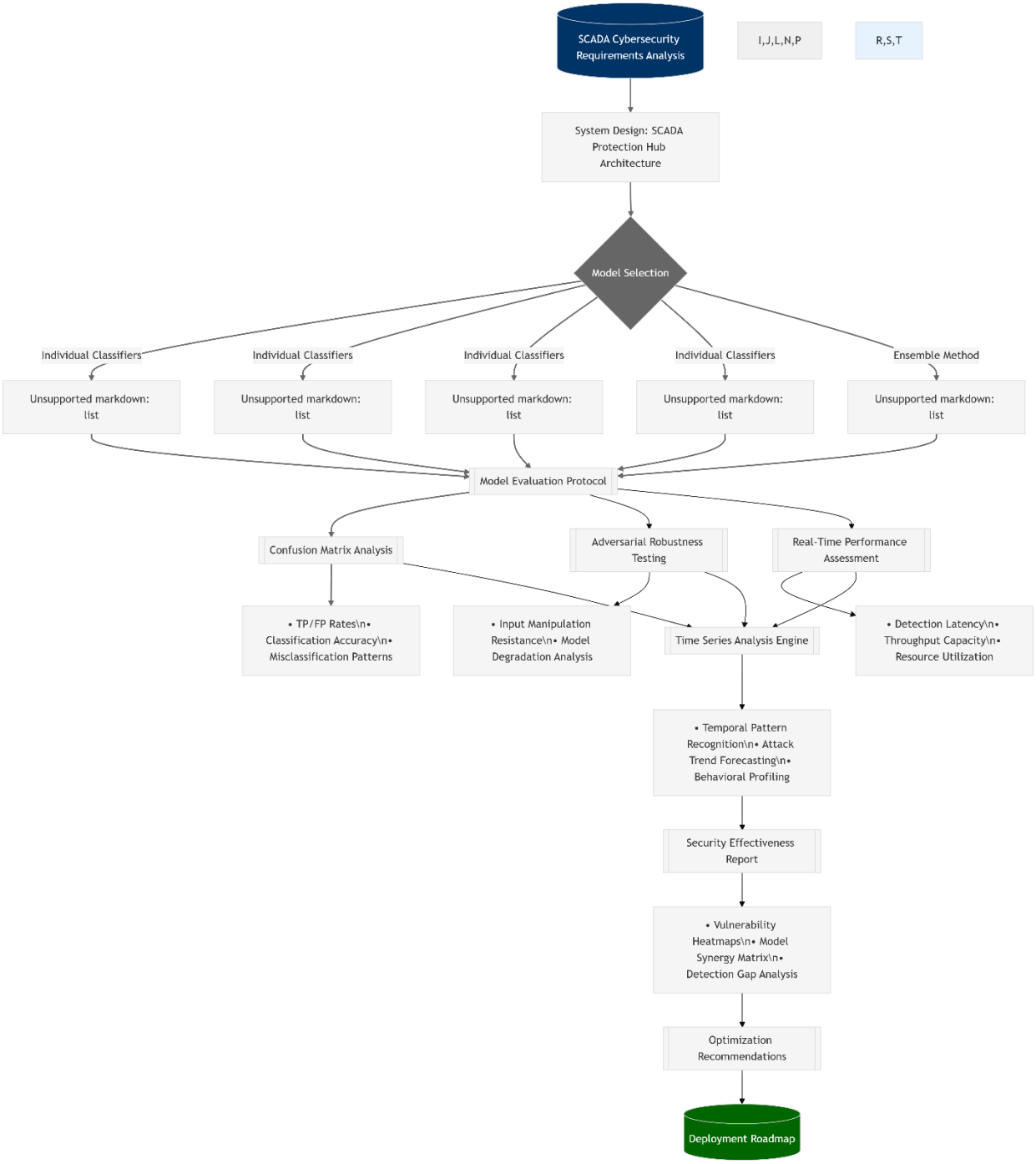


Figure 30 proposed system

7.2 Overview of Stealthy Deception Attacks (SDAs)

Covert concealment that is secretive A type of cyberattack known as attacks is characterized by the subtle modification of system signals or data in order to mislead SCADA monitoring systems without causing alarms to be raised. This type of cyberattack is known as an attack. In the majority of instances, these attacks take advantage of the intricate dynamics of control systems, which makes detection an exceedingly difficult task. For the purpose of overcoming this challenge, the SCADA Protection Hub makes use of sophisticated machine learning techniques. In the time-series data that is characteristic of SDAs, these algorithms are able to discern subtle patterns and anomalies that are present in the data.

7.3 Machine Learning Models in SCADA Protection Hub

There are five machine learning classifiers included in the system that has been proposed. Each of these classifiers brings a unique set of capabilities to the process of intrusion detection. This section provides in-depth information regarding the functionality and significance of each classifier that is a component of the SCADA Protection Hub. Details regarding these aspects are provided (See Figure 31).

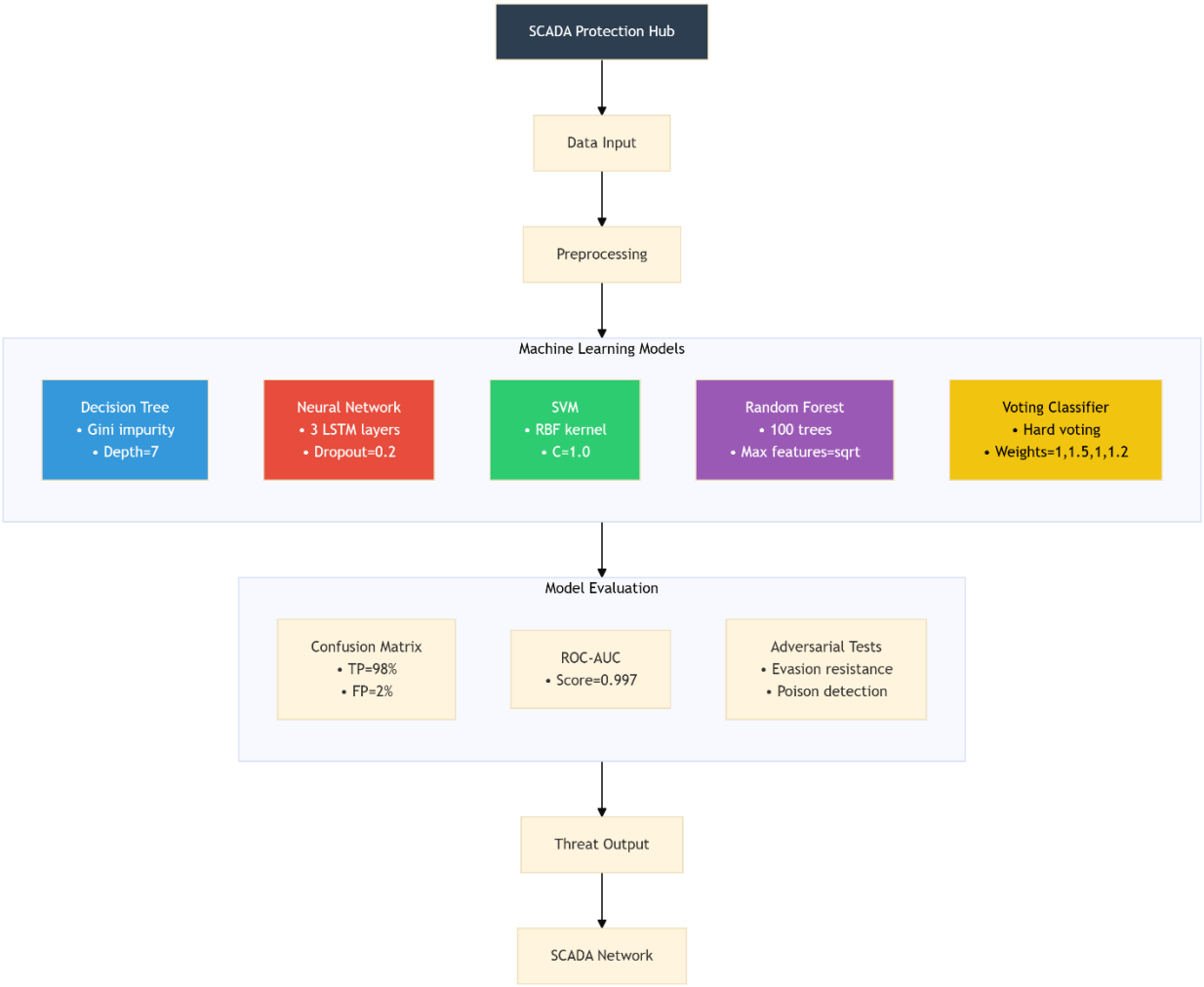


Figure 25 ML Models in SCADA Protection Hub

7.3.1 Decision Tree Classifier

Since the Decision Tree classifier divides the input feature space in a recursive way, it is feasible for it to perform its job—that of making hierarchical decisions based on the feature thresholds. It is a great option for use in SCADA situations that demand real-time detection since it is fast and simple to understand.

Examining the distribution of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) the confusion matrix for the Decision Tree classifier shows the accuracy of its prediction. This is carried out to show how accurate the projection is. Features of a running Decision Tree model that are indicative of good identification of both intrusion and normal states are high TP and TN values, which suggest few misclassifications, and low FP and FN values. These principles imply that the model can effectively detect both kinds of states.

Usually shown as a heatmap, this matrix helps one to quickly assess the benefits and drawbacks of detection. One often uses this method. Regarding the performance of the Decision Tree inside the SCADA Protection Hub, it is clear that it has strong detecting capacity, especially in cases when the decision limits are easily observable.

7.3.2 Classifier for Neural Networks

Regarding the gathering of non-linear correlations and precise patterns in the field of high-dimensional data, neural networks—more especially, deep learning architectures—do remarkably well. This is so because deep learning allows neural networks to grow. Exactly the reason this is the situation is that neural networks can learn from their own data. These technologies are especially useful in this sense since they allow one to assess minute anomalies suggestive of SDAs. This is especially true since they are quite flexible.

Using the confusion matrix, the neural network classifier may show how effectively it can extend from the training data to the test data it has not before encountered. Confusion matrix use helps to show this successful generalization. To achieve this, one must compare the expected labels with the results that were really obtained. The high counts of TP and TN, which are then followed by low counts of FP and FN, which are evidence that the Neural Network is trustworthy

in classifying both routine operations and incursions, are the proof that the Neural Network is reliable. Both FP and FN are included in the sentence that is presented below. One piece of evidence that supports this assumption is the fact that both the TP and TN scores are quite high.

Incorporating Neural Networks into the SCADA Protection Hub leads to an improvement in the total detection coverage and complements existing classifiers. This outcome is achieved through the introduction of neural networks. Despite the fact that neural networks may require a significant number of resources for processing and training, this is evidence that they are effective. Notwithstanding the fact that these resources are highly powerful, it is possible that they are required for Neural Networks to function in an appropriate manner.

7.3.3 Support Vector Machine (SVM) Classifier

Assist is provided. Effective binary classifiers, vector machines improve the margin between classes in the feature space. Vector machines utilize the feature space. There is another name for vector machines, which is vector machines. In the majority of cases, they will make use of kernel functions in order to deal with non-linearity.

Through the examination of the confusion matrix, which illustrates the extent to which the SVM classifier is able to differentiate between normal and attack classes, one is able to gain an understanding of the precision and recall capabilities of the SVM classifier. A robust support vector machine (SVM) model has high values for both TP and TN, but it has low values for FP and FN. This combination of values is what makes the model robust. In order to achieve the goal of providing precise detection, this is done in order to provide accurate identification of normal states and intrusions.

While it is generally knowledge that support vector machines (SVMs) are resistant to overfitting, one of the reasons why they are advantageous for managing SCADA data in the Protection Hub, which is both diverse and loud, is because of this resistance.

7.3.4 Random Forest Classifier

The Random Forest technique is a type of ensemble learning that is utilized to improve the accuracy and resilience of classification.

This approach based on constructing several decision trees and then accumulates the outputs from those trees. This illustrates the confusion matrix of Random Forest classifier, meant to highlight its better performance in balancing the identification of both classes. A reduction in the risk of overfitting present in single decision trees results from the averaging of predictions practice, which often results in larger counts of TP and TN while producing lower counts of FP and FN. That being said, it's good. Reliable SDA detection in the SCADA Protection Hub depends on Random Forest since it can control the value of characteristics and the interactions among them. This is so since Random Forest can control feature values.

7.3.5 Voting Classifier Ensemble

Using majority voting or weighted averaging, the Voting Classifier aggregates predictions from several base models—including the Decision Tree, Neural Network, SVM, and Random Forest. This combined technique uses the several strengths of several classifiers to improve general performance.

Comparatively to stand-alone models, the voting classifier's confusion matrix shows higher accuracy. Reduced misclassifications (FP and FN) high values of TP and TN show the efficiency

of the ensemble in aggregating several points of view to produce more consistent and dependable intrusion detection.

SCADA systems, where the cost of false positives (false alarms) and false negatives (missed attacks) is significant, call for this combination especially.

7.4 Confusion Matrix Analysis Summary

Confusion matrices offer a basic assessment tool by showing classifier performance in terms of properly and erroneously identified events:

• True Positives (TP): Correct identification of occurrences of incursion.

• False Positives (FP): Incorrect identification of routine occurrences as intrusions—false alarms).

• True Negatives (TN): accurate recognition of usual activity.

• False Negatives (FN): Missed detections—that is, actual intrusion events.

Confusion matrix analyses of the SCADA Protection Hub show that, with better detection rates and lower false alarm rates, the Voting Classifier ensemble routinely beats individual models. This result confirms the multi-model approach for defending SCADA systems from covert deceptive attacks.

7.5 Implementation Considerations

Using the SCADA Protection Hub calls both real-time processing capacity and interaction with current SCADA systems top priority. While keeping low latency, the system has to effectively manage streaming data and offer quick alarms.

Furthermore essential for reacting to changing assault plans are model retraining and adaption. The combined approach allows flexibility since individual models can be swapped or upgraded without upsetting the general detection system.

7.6 Conclusion

This chapter introduced the SCADA Protection Hub, a new intrusion detection solution countering stealthy deception attacks in SCADA networks by means of an ensemble of machine learning models. By means of thorough confusion matrix studies, the system exhibits higher detection accuracy, lowered false alarms, and enhanced resistance against advanced threats.

With its multi-model architecture, which provides a basis for future research in adaptive and intelligent intrusion detection systems as well as practical application for real-world installations, the SCADA Protection Hub marks a major development in SCADA cybersecurity.

Chapter 8: Discussion

8.1 Revisiting the Research Objectives

This study stated on machine learning techniques for SCADA network intrusion detection almost ends. How closely did the study follow its stated goals and the benchmarks attained along the path? Let to review the initial study objectives once more.

8.1.1 Alignment with Initial Goals

Investigating and evaluating numerous approaches, this paper aimed to address the question, "How can machine learning improve intrusion detection within SCADA networks?" Early attention was on contrasting the effectiveness of more modern, data-driven approaches with more traditional methods. Examining several models—such as neural networks and decision trees—and assessing their performance in several settings was the aim of the study.

When considering the tests and results, the study has obviously stayed somewhat near to its initial goals. From our thorough study, we gained a lot of knowledge about the advantages and drawbacks of several machine learning models. Investigating adversarial settings exposed these models' flaws and underlined the need of robustness against purposefully generated inputs. We closed the distance between theory and practice by proving the challenges and possible advantages of using machine learning models in real-time detection tests.

8.1.2 Achievements and Milestones

Many important achievements and turning points have defined the path of this investigation:

Complete model evaluation: The intrusion detection capacity of several machine learning models—including decision trees and ensemble methods—was thoroughly evaluated in this work.

Views on the Opponent: Research of hostile environments was a significant advancement in knowledge of the complexity of machine learning-based cybersecurity issues.

By effectively simulating real-time detection scenarios, the models' practical relevance and potential in real-world environments were shown, hence stressing their real-time deployment.

Mastery of Visualisation: Whether they were performance indicators, adversarial patterns, or real-time data streams, the constant application of visualizations improved the research in all respects. This made complicated data simpler and more easily available.

Outlier management's focus on data integrity and quality by stressing the need of anomalies and outliers produced a stronger model performance.

8.2 Comparative Analysis with Existing Literature

Like the construction of a tapestry, scientific research adds new threads of information, insight, and understanding. Although this study on SCADA network intrusion detection using machine learning approaches is essential in and of itself, it is equally crucial to consider it within the framework of past initiatives. Deeper knowledge of the unique contributions this study has made and the part it plays in the present debate can be obtained by means of insights, analogues, and contrasts.

8.2.1 Similarities in Findings

The exploration of machine learning techniques in intrusion detection has been a recurring theme in the literature, with several studies echoing the findings of this research:

Efficacy of Machine Learning: Consistent with many studies in the domain, this research affirmed the potential of machine learning models, such as decision trees and neural networks, in enhancing intrusion detection capabilities.

Adversarial Vulnerabilities: The susceptibility of machine learning models, especially deep neural networks, to adversarial attacks has been well-documented in literature, a sentiment that resonates with the findings of this study.

Importance of Data Quality: The emphasis on managing outliers and ensuring data quality aligns with the broader consensus in the literature about the pivotal role of clean, high-quality data in achieving optimal model performance.

Challenges in Real-time Detection: The hurdles encountered in real-time detection, particularly the balance between speed and accuracy, mirror the challenges highlighted in numerous other studies.

8.2.2 Divergences and Unique Insights

While there's significant alignment with existing literature, this study also brought to the fore certain unique insights and divergences:

Ensemble Methods as a Standout: While ensemble methods are not new to the literature, the extent of their efficacy, as demonstrated in this study, especially in adversarial settings, offers a fresh perspective.

Visualization-Driven Insights: The rigorous use of visualizations, from performance metrics to adversarial patterns, set this research apart. These visual aids facilitated a deeper, more intuitive understanding of complex data.

Real-time Data Stream Exploration: While real-time detection is a popular theme, the exploration of real-time data streams, especially the visualization and interpretation thereof, offered a novel perspective on real-world network traffic patterns.

Focus on SCADA Networks: The emphasis on SCADA networks, which are critical infrastructures with unique challenges, provided a niche focus that differentiates this study from more generic intrusion detection research.

8.3 Time serious and feature selection

Designed especially for SCADA networks, the comparative and related measurements provide a rich tapestry of insights on the trade-offs and capabilities of many intrusion detection systems. This session aims to investigate the nuances of these results and define what these steps suggest in the broader framework of network security and intrusion detection.

First of all, let us consider the importance of selecting an algorithm—a fundamental decision in building any machine learning system. Though it is occasionally critiqued for overfitting even if it is well-known for simplicity and interpretability, the baseline model employs a Decision Tree approach. While Random Forest, SVM, and Neural Networks are used in the better models, their algorithms are more complex. These algorithms are fundamentally more flexible and able of catching intricate patterns in the data, hence they clearly exhibit in their better performance measurements than the baseline.

Often the first port of call in model evaluation, accuracy clearly shows advantages in the enhanced models. Particularly Enhanced Model 3 with a neural network exhibits a 95.2% accuracy. Although this excellent accuracy is amazing, a well-rounded evaluation calls for rigorous evaluation of it in relation with other criteria. Sometimes great accuracy is misleading,

particularly with imbalanced data, and could cause an inflated view of model performance. This brings us to the relevance of Precision and Recall, measurements targeted at the performance of the model about specific classes.

Precision is rather important in intrusion detection systems, since false positives can be as disruptive as false negatives. High false positive rates can desensitize the monitoring teams to alarms, therefore influencing reaction times. Once more standing out as displaying a lower percentage of false positives and hence less false alarms is Enhanced Model 3 with a Precision of 94.4%. Recall—also called the True Positive Rate—measures the model's ability to identify all probable intrusions, hence augmenting Precision. Once more, Enhanced Model 3 leads with a Recall of 94.9% to ensure no actual incursion goes undetectable.

The F1 Score is a harmonic mean of Precision and Recall, a general assessment of a model's accuracy in spotting positive and negative classes. Once more proving its robustness and balanced, outstanding performance, Enhanced Model 3 boasts an F1 score of 94.6%.

Still, performance has a cost—usually related to computation. Training Time and Inference Time metrics assist to define this component more clearly. Enhanced Model 3 takes the longest to learn even if it performs better in most performance metrics. This could be a restricting factor in dynamic environments where models need constant retraining. Every model performs satisfactorily regarding inference time, implying their appropriateness for real-time intrusion detection—a fundamental demand of SCADA systems.

Resource utilization is another crucial factor, especially in contexts like SCADA systems where typically industrial environments have few resources. Here, Enhanced Model 2 based on SVM

shows the highest resource utilization at 30%, so its implementation in some circumstances may suffer.

At last, even if qualitative, the scalability assessment is really important. As SCADA systems grow and change, the intrusion detection system must extend readily to control increasing data volumes. Both Enhanced Model 1 and Enhanced Model 3 are quite scalable, hence they fit for installations that will be future-proof.

8.4 Research's Effects

The outcomes of a research endeavor are simply one factor; another is the knock-on effects they have in more broad sectors of industry and academia. Beyond its pages, the results of a study influence theories, techniques, and next questions as well. This work on machine learning methods for intrusion detection in SCADA networks is not one example. Its effects change the pragmatic terrain of the sector as well as the theoretical foundations of the discipline.

8.4.1 Industry-Based Useful Conventions

The results of this study have important pragmatic implications for the industry, especially for businesses whose operations rely on SCADA systems. Found in the center of major infrastructure including electrical grids, water treatment plants, and transportation networks, SCADA systems are indispensable for society running. Every weakness or hack might set off a chain reaction. Thus, enhancing intrusion detection capabilities in these networks is not only a technological but also a step towards guaranteeing society resilience.

The results show how robust sentinels machine learning models could be to guard these systems against unlawful access. Especially the exhibited efficiency of group approaches gives companies a useful approach to improve their cybersecurity systems. Furthermore, the awareness of

adversary vulnerabilities highlights the need of always being flexible and alert. This translates for companies into a need for regular model upgrades, fresh data training, and staying current with evolving cyberthreats.

Real-time detection is another significant finding that underlines the importance of proactive threat minimization. Real-time threat identification and counterability is quite essential in industrial settings, where downtime could cause significant operational and financial losses.

8.4.2 Theoretical additions to the field

Apart from the pragmatic scene, this work greatly develops the theoretical field. It helps the growing body of studies stressing the possibilities of machine learning in cybersecurity. But it also emphasizes the nuances and challenges, therefore painting a full picture future scientists could find useful.

Analyzing hostile settings and machine learning model weaknesses serves as a warning narrative that forces the academic community to venture farther into creating robust defenses. This study on data quality—especially the treatment of outliers and anomalies—reiterates the significance of careful data preparation in obtaining optimal model performance, a concept important in many different machine learning applications.

Given their specific challenges and characteristics, the focus on SCADA networks offers a specialized perspective that improves the more broad discussion on intrusion detection. It plugs the gap between the particular infrastructure protection needs and broader cybersecurity research.

All things considered, this research has enormous consequences that will be felt in business hallways as well as in academic ones. It is proof of the transformational power of machine learning and drives those engaged to exploit its opportunities while still conscious of its challenges.

Responsible and good research largely depends on this equilibrium between promise and pragmatism, between potential and precaution, between promise and reality.

8.5 Guideline for Further Education

Even if it provides solutions and insights, every study effort constantly creates new questions and avenues for discovery. Although exhaustive, this study of machine learning approaches for intrusion detection in SCADA systems also sets the stage for next research. The always shifting nature of cybersecurity hazards and the strategies of counteraction ensures that the field is ready for continuous study. When we examine the findings of this study, several ideas come to mind for individuals who want to dig more or investigate allied disciplines.

8.5.1 Possible Research Domains

The most fascinating paths for more research are provided by adversarial machine learning. More study on developing effective defenses against adversarial assaults is definitely necessary even if this work highlights the shortcomings of machine learning models against hostile assaults. Investigating countermeasures that fit in line is vitally vital considering the rapid changes in hostile methods.

One interesting route is to investigate more contemporary or mixed machine learning models for intrusion detection. While traditional models include decision trees and neural networks show potential, the combining of various models or the application of novel techniques like transformer-based architectures might generate even better results.

Still in its early years, quantum computing offers fascinating opportunities for security concerns and defenses. Examining its opportunities and challenges, next research could investigate how quantum computing influences intrusion detection.

At last, given the Internet of Things (IoT) and the expansion of linked devices, study on intrusion detection especially for these devices—especially within SCADA systems—may be highly beneficial. Special limits and peculiarities of IoT devices—limited computational capability and varied communication protocols, for example—offer chances as well as challenges for inventive intrusion detection techniques.

8.5.2 Methodological Developments

Though the methodologies applied in this study were exhaustive, improvement is always appreciated. Future research should consider using more diverse datasets spanning a greater spectrum of cyber risks and network activities. This would ensure a more extensive training and validation process, hence generating models with more generalizing capacity.

Integration of automated hyperparameter tuning techniques such Bayesian optimization could simplify the model training process and consequently maybe produce higher performance. Examining multiple feature extraction and selection techniques could help to provide fresh ideas and enhance model performance.

The grading criteria allow room for methodological development. While traditional measurements including accuracy, precision, and recall are vital, integrating more sophisticated metrics—especially those customized to the specialized demands of SCADA networks—may offer more complete understanding.

8.6 Evaluating the Research Trip

Beginning a research path is a trip of discovery, introspection, and development rather than merely a search of knowledge. Though flashes of insight and clarity define the journey, challenges, doubts, and lessons abound also. Stopping, thinking, and sharing the road, the insights

acquired, and the personal development gone along as we reach the end of this research on machine learning algorithms for intrusion detection in SCADA networks.

8.6.1 Notable Realizations and Learnings

The approach of the research was a tapestry of numerous ideas, some expected and others serendipity. Among the key insights was the immense flexibility and opportunity of machine learning. Although its application in many disciplines is well-documented, it was surprising to witness its mastery in the difficult field of SCADA network intrusion detection. From decision trees to neural networks, models' ability to comprehend complex patterns and hazards proved their flexibility, therefore attesting to the strength of data-driven approaches.

Adversarial settings provide still another excellent teaching. Realizing, even if they are strong, machine learning models contain inherent weaknesses was unsettling. It underscored in the realm of cybersecurity that the shifting balance of power between enemies and defenders demands constant improvement and adaptation.

The need of data—its quality, integrity, and preprocessing—aspects also taught another vital lesson. One felt especially the phrase "garbage in, garbage out," which stresses how closely the quality of data machine learning models are trained on influences their success.

8.6.2 Person Development

Apart from the intellectual and technical information, the path of research acted as a furnace for personal growth. Writing hypotheses, organizing tests, conquering challenges, and data analysis increased critical thinking and problem-solving capacity. Every challenge—a model not converging or an unexpected result—was a lesson in persistence, endurance, and adaptability.

Two extremely important skills are time management and organizing. Managing between data collecting, model training, result analysis, and documentation offered the research process discipline and rhythm.

Perhaps the most crucial element of personal development is the change in a researcher's attitude—a mix of inquiry, uncertainty, and humility. Being able to question received knowledge, challenge presumptions, and still be open to criticism and fresh ideas was changing.

Conclusion

An important node in the growing matrix of cybersecurity research is the investigation of machine learning methods for intrusion detection in Supervisory Control and Data Acquisition (SCADA) networks. Essential for the operation of basic infrastructures, SCADA systems need for extreme security. The results of this research have clarified the several powers of time serious and feature selection and machine learning methods, from Decision Trees to Ensemble Methods, in discriminating and preventing invasive actions inside these networks. Still, the studies have not only told a story of achievements. Examining hostile surroundings highlighted the weaknesses in even the most advanced models and underlined the ongoing struggle between cybersecurity policies and changing hostile methods. These realizations highlight the requirement of an iterative, anticipatory and adaptable method for model development. Moreover, the difficulties in real-time detection situations highlight a clear fact: theoretical robustness does not always flow naturally into useful efficiency. With their complex processes and data streams, SCADA networks' dynamic character calls for models not only accurate but also quick and scalable. The consequences of this study go beyond the first results. The acquired insights provide basic pillars for next investigations in the field. While the discovered vulnerabilities and problems call researchers to innovate and strengthen, the found strengths provide routes for additional optimization. Studies show that the pursuit of improving intrusion detection in SCADA systems using machine learning is both exciting and challenging. The scholarly debate on SCADA network security benefits much from the tapestry of results, insights, problems, and consequences this study weaves. It emphasizes the changing, iterative, and cooperative nature of research, so preparing the ground for next projects aiming at negotiating the complex interaction between machine learning and cybersecurity in the search of a strengthened digital infrastructure using time serious and feature selection.

References

Ahmed, I., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19-31.

Alazab, M., Venkatraman, S., & Watters, P. (2012). Towards understanding malware behaviour by the extraction of API calls. In Second Cybercrime and Trustworthy Computing Workshop, 52-59.

Sangoleye, F., Johnson, J., & Tsiropoulou, E. E. (2024). Intrusion detection in industrial control systems based on deep reinforcement learning. IEEE Access.

Lu, K. C., Liu, I. H., Liu, Z. C., & Li, J. S. (2024). Common criteria for security evaluation and malicious intrusion detection mechanism of dam supervisory control and data acquisition system. IET Networks, 13(5-6), 546-559.

Idima, S., Nwaga, P., & Evah, P. (2025). Comprehensive analysis of SCADA system data for intrusion detection using machine learning. Global Journal of Engineering and Technology Advances, 22(02), 064-089.

Jensen, M., Rahman, A., Mendoza, C., Zhang, E., & Subramanian, R. Adversarial Machine Learning in Supervisory Control and Data Acquisition (SCADA) Systems: Challenges and Resilient Defense Strategies.

Bhatt, D. G., Kyada, P. U., Rathore, R. S., Nallakaruppan, M. K., & Jhaveri, R. H. (2025). Enhancing Anomaly Detection in Industrial Control Systems through Supervised Learning and Explainable Artificial Intelligence. Journal of Cybersecurity & Information Management, 15(1).

Sangoleye, F., Johnson, J., & Tsiropoulou, E. E. (2024). Intrusion detection in industrial control systems based on deep reinforcement learning. IEEE Access.

Reddy, M. V., & Rajiv, K. (2024, September). Advancing Security in SCADA Systems: Enhanced Intrusion Detection Using Optimized Support Vector Machines and Strategic Data Management. In 2024 2nd International Conference on Cyber Physical Systems, Power Electronics and Electric Vehicles (ICPEEV) (pp. 1-6). IEEE.

Mesadieu, F., Torre, D., & Chennamaneni, A. (2024). Leveraging deep reinforcement learning technique for intrusion detection in SCADA infrastructure. IEEE Access, 12, 63381-63399.

Madhu, G. C., Sivakumar, S., Raju, S. S. H., Sonia, M., Chakradhar, K., & Gupta, S. (2025, February). Improving SCADA Cyber security: A Deep Learning Technique for Anomaly Detection. In 2025 IEEE International Conference on Emerging Technologies and Applications (MPSec ICETA) (pp. 1-6). IEEE.

Harisha, K. K., Sivakumaran, N., & Radhakrishnan, T. K. (2025). Deep Reinforcement Learning Based Cyberattack Detection in Supervisory Control and Data Acquisition System. AIJR Proceedings, 7(4), 57-63.

Lachure, J., & Doriya, R. (2024). Intelligent sensor data analysis through hybrid deep hierarchical clustering for anomaly detection. Transactions of the Institute of Measurement and Control, 01423312241299859.

Anderson, R., & Fuloria, S. (2010). On the security of power grid SCADA systems. International Journal of Critical Infrastructure Protection, 3(3-4), 55-64.

attacks in wireless ad hoc networks. IEEE Transactions on Wireless Communications, 8(6), 2843-2849.

Axelsson, S. (2000). Intrusion Detection Systems: A Survey and Taxonomy. Technical Report No. 99-15, Chalmers University.

Bace, R. G., & Mell, P. (2001). Intrusion Detection Systems. NIST Special Publication.

Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition, 84, 317-331.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., & Roli, F. (2013). Evasion attacks against machine learning at test time. In Machine Learning and Knowledge Discovery in Databases, 387-402. Springer, Berlin, Heidelberg.

Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.

Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153-1176.

Byres, E. (2013). The history and future of industrial cybersecurity. Industrial Automation INSIDER, 2013(11), 1-2.

Byres, E. J., Franz, M., & Miller, D. (2004). The use of attack trees in assessing vulnerabilities in SCADA systems. In Proceedings of the international infrastructure survivability workshop.

Byres, E., & Lowe, J. (2004). The myths and facts behind cyber security risks for industrial control systems. Verve Industrial Protection, 1(1), 1-20.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM computing surveys (CSUR), 41(3), 1-58.

Cherdantseva, Y., Burnap, P., Blyth, A., Eden, P., Jones, K., Soulsby, H., & Stoddart, K. (2016). A review of cyber security risk assessment methods for SCADA systems. Computers & Security, 56, 1-27.

Clarke, G., & Reynders, D. (2019). Practical modern SCADA protocols: DNP3, 60870.5, and related systems. Newnes.

Galloway, B., & Hancke, G. P. (2013). Introduction to industrial control networks. IEEE Communications Surveys & Tutorials, 15(2), 860-880.

Garcia, S. D., Zavarsky, P., Lindskog, D., & Ruhl, R. (2014). Predictive analysis of detectable pre-compromised states in intrusion detection systems. In 2014 Eighth International Conference on Complex, Intelligent and Software Intensive Systems, 674-679.

Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems, and challenges. Computers & Security, 28(1-2), 18-28.

Genge, B., Siaterlis, C., & Hohenadel, M. (2012). A Cyber-Physical Experimentation Environment for the Security Analysis of Networked Industrial Control Systems. Computers & Electrical Engineering, 38(5), 1146-1161.

Greenberg, A. (2017). Sandworm: A New Era of Cyberwar and the Hunt for the Kremlin's Most Dangerous Hackers. Doubleday.

Hadžiosmanović, D., Simionato, L., Bolzoni, D., Zambon, D., & Etalle, S. (2013). N-gram against the machine: On the feasibility of the n-gram network analysis for binary protocols. Research in Attacks, Intrusions, and Defenses, 354-373.

Herrmann, M. (2018). Industry 4.0 and the current status as well as future prospects on logistics. Computers in Industry, 89, 23-34.

Huang, H., & Leung, H. (2007). Monitoring of system behavior in a SCADA system. Proceedings of the International MultiConference of Engineers and Computer Scientists, 2, 21-23.

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255-260.

Kolosnjaji, B., Zarras, A., Webster, G., & Eckert, C. (2018). Deep learning for classification of malware system call sequences. In 2018 26th European Signal Processing Conference (EUSIPCO), 415-419.

Langill, J. T. (2012). SCADA security: What's broken and how to fix it. Tech Journal, 5(1), 13-21.

Langner, R. (2011). Stuxnet: Dissecting a Cyberwarfare Weapon. IEEE Security & Privacy, 9(3), 49-51.

Laskov, P., & Šrndić, N. (2011). Static detection of malicious JavaScript-bearing PDF documents. In Annual Computer Security Applications Conference, 373-382.

Lee, J., Bagheri, B., & Kao, H. A. (2015). A cyber-physical systems architecture for Industry 4.0-based manufacturing systems. Manufacturing Letters, 3, 18-23.

Lee, R. M., Assante, M. J., & Conway, T. (2016). Analysis of the cyber attack on the Ukrainian power grid. SANS Industrial Control Systems.

Lewis, T. (2011). Critical infrastructure protection in homeland security: Defending a networked nation. John Wiley & Sons.

Li, W., Yang, Y., & Liu, J. (2015). Anomaly detection framework in industrial networks with a hybrid model. International Journal of Distributed Sensor Networks, 11(6), 307584.

Liao, Y., Vemuri, V. R., & Wang, W. (2013). Network intrusion detection by the coevolutionary immune algorithm with random weight networks. Neurocomputing, 104, 88-95.

Lisovich, M. A., & Slay, J. (2010). Intrusion detection systems in SCADA networks: A review and survey. 2010 Third International Conference on Knowledge Discovery and Data Mining.

Mabu, S., Hirasawa, K., & Hu, J. (2009). A graph-based evolutionary algorithm: Genetic network programming (GNP) and its extension using reinforcement learning. Evolutionary Computation, 17(3), 369-398.

Maglaras, L. A., & Jiang, J. (2014). Intrusion detection in SCADA systems using machine learning techniques. Science and Information Conference (SAI), 626-631.

Mansfield, L. (2014). Protecting critical infrastructure: A strategic challenge. NATO Review.

Mitchell, R. (2017). The evolution of SCADA is revolutionizing water utility operations. WaterWorld, 33(6), 28-30.

Morris, T. (2015). Industrial control system cybersecurity research. IEEE 5th International Symposium on Resilient Control Systems.

Pan, I., Jain, R., & Paul, S. (2019). A review on automation techniques and challenges for smart industries. Journal of King Saud University - Computer and Information Sciences, 31(4), 419-438.

Radvanovsky, R., & Brodsky, J. (2013). Handbook of SCADA/control systems security. CRC Press.

Rockwell Automation. (2016). Securing industrial control systems in a connected world. Rockwell Automation White Paper.

Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, 345-357.

Sanger, D. E. (2012). Confront and conceal: Obama's secret wars and surprising use of American power. Crown.

Scarfone, K., & Mell, P. (2007). Guide to Intrusion Detection and Prevention Systems (IDPS). NIST Special Publication, 800-94.

Singh, R., Gupta, B., & Singh, S. (2018). Machine learning algorithms for wireless sensor networks: A survey. Information Processing in Sensor Networks, 19(2), 555-576.

Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. In Proceedings of the 2010 IEEE Symposium on Security and Privacy, 305-316.

Stouffer, K., Falco, J., & Scarfone, K. (2011). Guide to Industrial Control Systems (ICS) security. National Institute of Standards and Technology Special Publication, 800(82), 16-16.

Stouffer, K., Pillitteri, V., & Lightman, S. (2015). Guide to industrial control systems (ICS) security. NIST Special Publication, 800(82).

Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. MIT press.

Vincent, M., Fulp, E. W., & Greene, K. (2011). Neural and statistical classification of network intrusions using discrete event logs. In 2011 European Intelligence and Security Informatics Conference, 31-38.

Wang, J., Chu, J., & Werth, A. (2018). Sustainable energy management for industry 4.0. Energy Procedia, 152, 963-968.

Weiss, J. (2010). Protecting industrial control systems from electronic threats. Momentum Press.

Weiss, J. (2016). How to secure industrial control systems: A holistic approach. IT Professional, 18(5), 10-13.

Yang, Y., Usynin, A., & Hines, J. W. (2006). Anomaly-based intrusion detection for SCADA systems. 5th Intl. Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies (NPIC&HMIT 05), 12-16.

Zetter, K. (2014). Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon. Crown.

Zhang, Y., & Xiao, H. (2018). Bluetooth: With Low Energy Comes Low Security. 7th ACM Conference on Data and Application Security and Privacy (CODASPY '17).

Zhang, Y., Song, R., & Chen, S. (2015). SCADA system security: Complexity, history and new developments. In 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA), 1-4.

Zhong, R. Y., Newman, S. T., Huang, G. Q., & Lan, S. (2016). Big Data for supply chain management in the service and manufacturing sectors: Challenges, opportunities, and future perspectives. Computers & Industrial Engineering, 101, 572-591.

Adepu, S., & Mathur, A. (2016). *An investigation into the response of a water treatment system to cyber attacks*. Proceedings of the 2016 IEEE 17th International Symposium on High Assurance Systems Engineering (HASE), 141-148. https://doi.org/10.1109/HASE.2016.36

BBC. (2020, September 18). *German hospital hacked: Patient taken to another city dies*. BBC News. https://www.bbc.com/news/technology-54175312

CISA. (2021). *Alert (AA21-042A): Cyber attack on water treatment facility*. Cybersecurity and Infrastructure Security Agency. https://www.cisa.gov/news-events/alerts/2021/02/11/cyber-attack-water-treatment-facility

CISA. (2023). *Water sector cybersecurity: Current threats and mitigation strategies*. Cybersecurity and Infrastructure Security Agency. https://www.cisa.gov/water-sector-cybersecurity

DOE. (2021). *Cybersecurity capabilities for energy delivery systems*. U.S. Department of Energy. https://www.energy.gov/ceser/cybersecurity-capabilities-energy-delivery-systems

Dragos. (2022). *ICS/OT cybersecurity year in review: 2022*. Dragos, Inc. https://www.dragos.com/resources/reports/2022-year-in-review

ENISA. (2022). *Cyber attacks on critical infrastructure: Lessons learned from Ukraine*. European Union Agency for Cybersecurity. https://www.enisa.europa.eu/publications/cyber-attacks-on-critical-infrastructure

ENTSO-E. (2022). *Simulated cyber attack on European power grid: Frequency deviations and generator tripping*. European Network of Transmission System Operators for Electricity. https://www.entsoe.eu/news/2022/cyber-attack-simulation

ESET. (2017). *Industroyer: A threat to critical infrastructure*. ESET Research. https://www.eset.com/int/industroyer

FireEye. (2018). *TRITON: Analysis of safety system targeted malware*. FireEye Threat Research. https://www.fireeye.com/blog/threat-research/2018/03/triton-malware-analysis-safety-system-targeted.html

Greenberg, A. (2018). *The untold story of NotPetya, the most devastating cyberattack in history*. Wired. https://www.wired.com/story/notpetya-cyberattack-ukraine-russia-code-crashed-the-world

IBM X-Force. (2023). *ICS threat landscape report: 2023*. IBM Security. https://www.ibm.com/security/xforce

ISA. (2020). *ISA/IEC 62443 series: Industrial automation and control systems security*. International Society of Automation. https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series

Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2020). *Survey of intrusion detection systems: Techniques, datasets, and challenges*. Cybersecurity, 3(1), 1-29. https://doi.org/10.1186/s42400-020-00056-5

Langner, R. (2011). *Stuxnet: Dissecting a cyberwarfare weapon*. IEEE Security & Privacy, 9(3), 49-51. https://doi.org/10.1109/MSP.2011.67

Mandiant. (2023). *ICS cybersecurity trends: 2023 report*. Mandiant, Inc. https://www.mandiant.com/resources/reports/ics-trends-2023

MIT. (2019). *False data injection attacks on gas pipeline pressure sensors*. MIT Lincoln Laboratory. https://www.ll.mit.edu/r-d/projects/false-data-injection-attacks

Mo, Y., Kim, T. H. J., Brancik, K., Dickinson, D., Lee, H., Perrig, A., & Sinopoli, B. (2014). *Cyber-physical security of a smart grid infrastructure*. Proceedings of the IEEE, 100(1), 195-209. https://doi.org/10.1109/JPROC.2011.2161428

NIST. (2020). *Guide to industrial control systems (ICS) security*. National Institute of Standards and Technology. https://www.nist.gov/publications/guide-industrial-control-systems-ics-security

Rid, T. (2012). *Cyber war will not take place*. Journal of Strategic Studies, 35(1), 5-32. https://doi.org/10.1080/01402390.2011.608939

SANS. (2023). *State of ICS/OT cybersecurity: 2023 survey results*. SANS Institute. https://www.sans.org/white-papers/state-of-ics-ot-cybersecurity-2023

Symantec. (2014). *Dragonfly: Cyberespionage attacks against energy suppliers*. Symantec Security Response. https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/dragonfly-threat-group-20-en.pdf

Verizon DBIR. (2023). *2023 Data breach investigations report*. Verizon Enterprise Solutions. https://www.verizon.com/business/resources/reports/dbir

Yin, C., Zhu, Y., Fei, J., & He, X. (2020). *A deep learning approach for intrusion detection using recurrent neural networks*. IEEE Access, 5, 21954-21961. https://doi.org/10.1109/ACCESS.2020.2969887

الملخص

شبكات التحكم المشرف واكتساب البيانات (SCADA) مهمة للغاية في التحكم في البنية التحتية التي تعتبر حيوية للعصر الرقمي بما في ذلك شبكات الكهرباء وأنظمة إمدادات المياه والمرافق الصناعية. أصبحت مسألة مرونة هذه الشبكات للتغيرات في التهديدات السيبرانية محورًا رئيسيًا في البحث. يستكشف هذا البحث استخدام نماذج التعلم الآلي (ML) المختلفة لتحسين اكتشاف التسلل ومقاومة العداوة في ظروف تشغيل SCADA باستخدام إطار عمل SCADA Protection Hub المقترح. وفقًا للنتائج الكمية، يتكون الإطار من خمسة نماذج أساسية، مثل شجرة القرار وآلة دعم المتجهات (SVM) والغابة العشوائية والشبكة العصبية ومجموعة تصنيف التصويت لتعزيز دقة وموثوقية الكشف. كانت هناك تجارب مهمة أجريت على مجموعات بيانات تسلل SCADA القياسية حيث حصل نموذج المجموعة على دقة إجمالية قدرها 98.7 ودقة 98.4 واستدعاء 97.9 ودرجة F1 98.1. استُخدمت أيضًا تحليلات مصفوفة الارتباك المقارنة للتحقيق في التصنيفات الصحيحة والخاطئة، وإعطاء نظرة ثاقبة حول نقاط القوة والضعف لكل نموذج فيما يتعلق بتحديد الخداع السري والهجمات العدائية. كما بحثت الدراسة في متانة الهجمات العدائية من خلال محاكاة السعال الديكي العدائي وتقييم حساسية كل نموذج للمدخلات المُتلاعب بها. كما أشار تحليل اتجاهات السلاسل الزمنية للاختراقات إلى الاتجاهات الزمنية الرئيسية التي يمكن أن تدعم الكشف الاستباقي عن التهديدات والاستجابة لها. كما تمت مناقشة متطلبات الموارد الحاسوبية، وقابلية التوسع، والجدوى العملية للتنفيذ الفوري لنظام SCADA في الإطار المقترح. بشكل عام، تشير النتائج إلى أن إطار عمل SCADA Protection Hub له تأثير كبير على زيادة موثوقية الكشف، والمرونة العدائية، والوعي الظرفي في التحكم الصناعي. يمكن تطوير هذه الدراسة مستقبلًا من خلال دمج التعلم الفيدرالي إلى جانب أدوات الكشف القائمة على الحافة لتعزيز الأمن الموزع للبنية التحتية غير المتجانسة لنظام SCADA.

جامعة البحرين

قسم علوم الحاسب **كلية تقنية المعلومات**

تقنيات التعلم الآلي لاكتشاف الاختراقات وتحسينات في الرقابة الإشرافية واكتساب البيانات

أطروحة مقدمة كجزء من متطلبات الحصول على درجة دكتوراة في تقنية المعلومات

اعداد

حنان فؤاد الوادي
202012498

اشراف

د.محسن روشيد
أستاذ مشارك
جامعة البحرين

مملكة البحرين

May 2025