2021 8th International Conference on Internet of Things: Systems, Management and Security (IOTSMS) | 978-1-6654-5868-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/IOTSMS53705.2021.9704986

On the Performance of Isolation Forest and Multi Layer Perceptron for Anomaly Detection in Industrial Control Systems Networks

Saja Alqurashi
Computer Science Department
Colorado State University
Fort Collins, CO USA
saja.alqurashi@colostate.edu

Hossein Shirazi
Computer Science Department
Colorado State University
Fort Collins, CO USA
shirazi@colostate.edu

Indrakshi Ray
Computer Science Department
Colorado State University
Fort Collins, CO USA
Indrakshi.Ray@colostate.edu

***Abstract*—With an increasing number of adversarial attacks against Industrial Control Systems (ICS) networks, enhancing the security of such systems is invaluable. Although attack prevention strategies are often in place, protecting against all attacks, especially zero-day attacks, is becoming impossible. Intrusion Detection Systems (IDS) are needed to detect such attacks promptly. Machine learning-based detection systems, especially deep learning algorithms, have shown promising results and outperformed other approaches. In this paper, we study the efficacy of a deep learning approach, namely, Multi Layer Perceptron (MLP), in detecting abnormal behaviors in ICS network traffic. We focus on very common reconnaissance attacks in ICS networks. In such attacks, the adversary focuses on gathering information about the targeted network. To evaluate our approach, we compare MLP with isolation Forest (iForest), a statistical machine learning approach. Our proposed deep learning approach achieves an accuracy of more than 99% while iForest achieves only 75%. This helps to reinforce the promise of using deep learning techniques for anomaly detection.**

***Index Terms*—ICS · SCADA · Reconnaissance Attack · Deep Neural Networks · MLP · iForest**

1. Introduction

Industrial Control Systems (ICSs) often form a nation's critical infrastructure and have become the target for attacks by hostile governments and terrorist groups. The effect ranges from operational disruption to physical damage and even leads to the loss of human lives. Examples include Stuxnet worm that damaged the uranium enrichment facility at Natanz, Iran in 2010 and Shamoon virus that attacked the oil company Aramco in 2012 [1].

ICSs were originally designed for operating in an isolated environment and security was not given much consideration. However, the connectivity between the ICS and Information Technology (IT) networks has increased the attack surface and makes such systems more vulnerable. The security solutions developed for IT systems cannot be used in ICS for various reasons. First, the priorities for ICS designers and IT developers are different. While confidentiality and integrity are essential in IT systems, functionality and availability are more critical to ICS. Second, ICS consist of proprietary and legacy components having long life-spans with poor in-built security in contrast to IT systems where the components are upgraded relatively frequently. Third, the supply chain for ICS systems has many different vendors each with its own security practices; ensuring security in a consistent manner across such a diverse landscape is more challenging compared to traditional IT systems having fewer players. Fourth, ICS components are often resource-constrained, which makes running malware or anti-virus software infeasible with delayed security patching. As the existing security defenses developed for IT systems cannot be used, they must be adapted or new ones developed for ICS.

Problem Statement. In this study, we want to detect anomalies that happen in the ICS network due to the adversary's activities. We scope our work only to one type of attack, called reconnaissance attacks. Specifically, we want to detect abnormal behaviors in the ICS networks by capturing packets of the network solely. The input of the proposed approach would be network packets in terms of feature value vectors. The proposed approach will answer if the given packets belong to the normal behavior of the system or are suspected to be abnormal behavior. In addition, a one-time auxiliary labeled input will be given to the proposed approach so it can learn the behavior of ICS in normal situations and abnormal incidents.

Proposed Approach. In this study, we propose a machine learning-based system that can detect abnormal behaviors and map them to the attacks against the system. Using statistical machine learning approaches is not ideal as they fail to capture the complex structure in the data. To address this limitation, we use a deep learning network and compare the results with statistical approaches. We explore the viability of deep learning in detecting reconnaissance attacks, including a port and address scanner and a device identification attack.

Key Contributions. Our major contributions in this paper are as follows. We propose two types of machine learning-based anomaly defections namely iForest and deep learning-based Multi Layer Perceptron (MLP). We test the two algo-

rithms against reconnaissance attacks, which are the most common attacks in ICS network. Our results reveal that (i) deep learning-based MLP algorithms can detect abnormal behavior precisely with more than 99% accuracy, and (ii) deep learning MLP significantly outperformed well-known iForest algorithm.

The remainder of the paper is organized as follows: Section 2 presents a background on ICS, reconnaissance attacks and, existing machine learning-based abnormal behavior detection studies. Section 3 describes our approach, the dataset that we use, and the machine learning metrics that we use for evaluation. Section 4 discusses the experiments we use to evaluate the proposed model and our results. Section 5 concludes the paper with paths for future work.

2. Background ad Related Work

In this section, we first briefly explain the ICS reference model and then explain the reconnaissance attack against ICS networks.

2.1. ICS Reference Model

We start by describing the ICS Reference Model. ICS has three main components: (i) programmable logic controllers (PLCs) and remote terminal unit (RTU) which form the control layer in the ICS, (ii) field devices that form the sensing device layer, and (iii) the devices through which Human-Machine Interactions (HMIs) occur which forms the human interface Layer [2]. Supervisory Control and Data Acquisition (SCADA) is a system to manage the ICS network. It provides a graphical user interface for operators to easily observe the status of a system, receive any alarms indicating out-of-band operation, or enter system adjustments to manage the process under control [2]. Field devices that include sensors and actuators are located underneath these layers. Figure 1 depicts these levels.

Level 0 involves the field devices that include sensors and actuators and are directly connected to the physical processes. This level is called Input and Output (I/O) Network.

Level 1 is a control network that involves the function of sensing and manipulating physical processes such as receiving and processing the data and triggering outputs, which are all done using PLCs.

Level 2 is supervisory control local area network. This network is responsible for monitoring and controlling the physical processes and the general deployment of systems such as workstations and history logs.

Level 3 is an IT environment, which is responsible for operations such as file transfer, hosting the websites, interacting with the mail servers and the cloud. This level is called the corporate network.

Our work is focused on attacks at Levels 1 and 2.

2.2. Reconnaissance Attack

The most common attacks in an ICS network are reconnaissance attacks [3, 4]. In this attack, the adversary gathers information via network probing, social engineering, and physical surveillance about the network. Mathur and Tippenhauer [5] showed reconnaissance attacks usually occur at level 2 and level 1 to gather information about PLC communications.

Teixeira et al. [2] gathered four types of reconnaissance attacks carried out against their test-bed including port scanners, address scan, device identification (in regular and aggressive modes), and exploit. Table 1 summarizes these four types.

These attacks differ with respect to the vulnerabilities they exploit, the mechanisms they deploy, and their impact. However, all these attacks will show some form of abnormal behavior. If these abnormalities can be detected promptly, they may prevent damages.

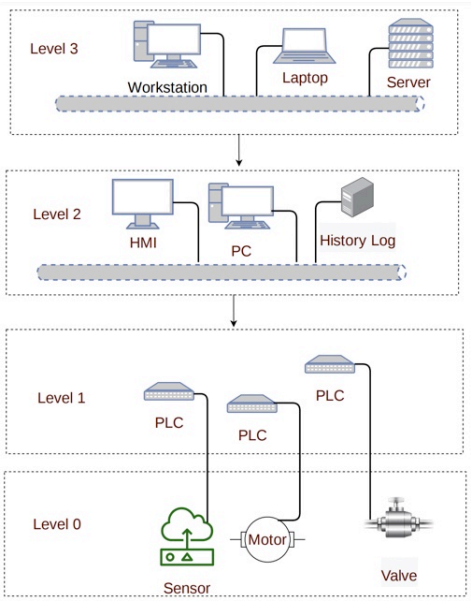

Figure 1. ICS reference model[2]

TABLE 1. RECONNAISSANCE ATTACKS CARRIED OUT AGAINST [2] TEST-BED

Attack Name	Description
Port Scanner	Identifying common SCADA protocols on the network using Nmap tool
Address Scan	Scanning network addresses and identifying the Modbus server address
Device Identification	Enumerating the SCADA Modbus slave IDs on the network and collecting additional information
Exploit	Reading the status of SCADA device statuses controlled by the PLC sensors

2.3. Detection Approaches

In this section, we describe few machine learning approaches that have been used for detecting anomalies in the

ICS networks.

One-Class Support Vector Machine (OCSVM) has been researched in depth to detect the abnormal behaviors in networked control systems, both in the academia and in the industry [6]. For example, Schuster et al. [7] used OCSVM to construct a self-configuring algorithm on several real-world industrial traffic traces. OCSVM is trained on normal traffic and can detect abnormal behavior. It has a one-class classification mechanism that can detect outliers from normal traffic. These outliers are considered to be abnormal.

Isolation Forest (iForest) algorithm has been widely used in recent researches for anomaly detection [8, 9, 10]. For example, Xu et al. [8] proposed an anomaly detection algorithm referred to as SA-iForest. This method uses a Simulated Annealing algorithm to optimize iForest. The use of Simulated Annealing improves the accuracy, reduces the computational complexity, and generalizes the iForest-based anomaly detection. iForest is sensitive only to the global outliers and is unable to detect local outliers. To address this limitation, Cheng et al. [10] combined iForest with Local Outlier Factor (LOF) to create a hybrid approach. The proposed solution reduces the time complexity of the algorithm by pruning out normal data points and generates outlier candidates for the next step.

Deep learning algorithms have been widely used to detect abnormal behaviors in ICS networks as statistical classifier techniques are not ideal when it comes to high dimensional data having complex structures such as those present in network traffic [7, 11, 12]. Deep learning algorithms can address this challenge as they can model complex behaviors of such systems through complex network architecture.

Wang et al. [13] used the autoencoder model to learn the normal behavior of devices. This model makes prediction and reconstitution of the input data simultaneously, which is different from the common autoencoder neural network. The exponentially weighted moving average method (EWMA) was used to calculate the smoothing error for the normal data set and then used as a threshold for detecting anomalies. The authors evaluated their proposed approach on the SWaT dataset with 88.5% recall and 87.0% F1-score.

Current research mostly focuses only on the protocol header fields and does not consider low-performance field devices. However, those devices are vulnerable to threats as well. Kim et al. [14] categorized the ICS reference model into two levels, namely, operative and product process management levels. Their autoencoder-based anomaly detection approach is a fast and lightweight algorithm that can be used for low-performance devices and can detect anomalies in real-time. In this approach, the autoencoder network was trained with the normal behavior of devices and abnormal behaviors were detected based on the reconstruction error, as the difference between the input and output of the autoencoder network. The approach has been tested on the SWaT dataset.

2.4. ICS Datasets

Detecting anomalies in the ICS networks depend widely on the availability of the data. Certain parameters like the existence of attack samples in the dataset and having well-balanced datasets are also important. Gómez et al. [15] proposed a methodology to generate reliable datasets for ICS systems that address issues related to data collection in ICS networks. Authors defined four main steps of *attack selection*, *attack deployment*, *traffic capture*, and *feature computation*. The authors also published a dataset called Electra, based on a railway electric traction substation. Mathur and Tippenhauer [5] published the Secure Water Treatment (SWaT) which is a reference dataset. The dataset was collected from a real water treatment test-bed on 7 days of normal activity and 4 days of data injection. Similarly, the Water Distribution (WADI) dataset [16] contains 16 days of logs of 123 industrial sensors and actuators. 15 attacks have been launched over 2 days of log capturing. For our experiments in this study, we use SWaT dataset [15]. We will use WADI dataset [16] as part of our future work.

3. Proposed Approach

In this section, we explain our detection approaches. We implement an unsupervised algorithm, namely, iForest, and a supervised deep-learning binary classification algorithm, namely, Multi Layer Perceptron (MLP) algorithm. We then describe the dataset that we use and also our evaluation metrics.

3.1. Machine Learning-based detection approaches

In our study we examine the two approaches on ICS networks data set, and evaluate and compare the accuracy of both approaches. First we examine the iForest which is a statistical approach. In the second experiment we examine MLP which is a deep learning approach.

Anomaly detection using machine learning in general has two phases:

- **Training phase**: building a model based on a training data set.
- **Testing phase**: each instance in the test set is passed through the model that was built in the previous stage, and a proper "anomaly score" is assigned to the instance.

iForest algorithm has been successfully applied for real-world anomaly detection applications. iForest algorithm is an unsupervised machine learning algorithm that is built based on the decision tree theory. Unlike model-based algorithms, e.g., statistical methods, classification-based methods, and clustering-based methods, that profile regular behaviors of systems, iForest isolates anomalies instead [17]. Two major drawbacks of these model-based anomaly detection systems are (i) being optimized to profile normal instances, not detecting abnormal instances, and (ii) are constrained to low volume of datasets and small dimension

TABLE 2. MAIN FEATURES SWAT DATASET WE USED IN OUR EXPERIMENTS

Feature	Description
Port	Port number of the source
Total packets	Total transaction packet count
Total bytes	Total transaction bytes
Source packets	Source/destination packet count
Destination packets	Destination packet count
Source bytes	Transaction bytes

size. iForest addresses both these issues by focusing on anomalies and isolating those rather than profiling normal behavior. The method is based on two properties of abnormal instances: (i) there are fewer number of abnormal instances and (ii) abnormal instances have attributes with values very different from normal instances. The authors created a more effective tree to isolate every single instance. Abnormal instances are isolated closer to the root while normal instances are isolated at the deeper nodes of trees. The algorithm achieves a low linear time-complexity and a small memory requirement which fits better for light-weight devices such as those found in ICS.

For implementation, we used Python3 with scikit-learn package [18] and iForest library. In our experiment, we used 3102 total nodes in 1000 trees.

MLP Deep Learning In addition to iForest, we examine Multi-Layer Perceptron (MLP) for binary classification. The algorithm is a supervised learning, meaning needs labeled samples, i.e., each sample need to be specified if it is normal or abnormal. MLP is a type of feed-forward artificial neural network (ANN) with multiple layers of perceptrons with an activation function. MLP involves at least three layers which are the input layer, hidden layer, and output layer [19]. When MLP algorithms are applied to supervised learning, the algorithm is trained on a set of input-output pairs and learns dependencies between inputs and outputs. During the training phase, the algorithm is adjusting parameters based on the error rate until it does not improve any further. In this work, the network is implemented with three layers, with 13 nodes in each hidden layer. We used Adam as the optimizer and ReLu as the activation function.

3.2. Dataset

In this study, we use the SWaT dataset which is one of the first ICS datasets that has been made publicly available. This dataset closely matches real-world industrial systems and emulates realistic cyber-attacks. This dataset has 7 million instances. We use 70% of data (5.2 million instances) for training and reserve 30% for testing (1.7 million instances). Table 3.2 lists the main features of this dataset we use in our experiments.

TABLE 3. DEFINITION OF METRICS

Data Class	Description
TN	*Normal* samples classified as *normal*
FN	*Normal* samples classified as *abnormal*
FP	*Abnormal* samples classified as *normal*
TP	*Abnormal* samples classified as *abnormal*

3.3. Evaluation Metrics

For comparing results, we use the following machine learning metrics.

$$Recall = \frac{TP}{TN + TP} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TN + TP + NF + TN} \tag{3}$$

$$F1 = 2 * \frac{precision * recall}{precision + recall} \tag{4}$$

We have an imbalanced dataset as the number of normal samples is significantly more than attack samples. Thus, we report both precision and F1 scores.

4. Results and Discussion

In this section, we report our results from conducting experiments on the SWaT dataset on two algorithms, namely, iForest and binary classification MLP algorithm.

Table 4 reports accuracy and F-1 scores for the two algorithms. The deep learning algorithm shows high accuracy and F-1 score, both 99%. On the other hand, the iForest algorithm has shown 75% accuracy that is significantly less than the deep learning model. The results show that the deep neural networks approach can detect reconnaissance attacks better than the iForest algorithm. We are concerned about the high positive rate with a low false-negative rate in any anomaly detection system; otherwise, false alarms or undetected abnormalities will degrade the usefulness of the system and endanger the safety of users.

TABLE 4. RESULTS FROM CONDUCTING EXPERIMENTS ON SWAT DATASETS WITH TWO DIFFERENT ALGORITHMS

Algorithm	Precision	F-1
iForest	75%	90%
Deep Learning	99%	99%

Table 5 compares the result of our deep learning approach (MLP) with the state-of-the-art solutions for the SWaT dataset. Our proposed MLP algorithm outperformed

3 other algorithms existing in the literature. Our algorithm has 99% precision and F1-score which is more than any other algorithm

TABLE 5. STATE-OF-THE-ART COMPARISON (SWAT DATASET)

Algorithm	Precision	F-1
MLP	99%	99%
DNN [20]	91.85%	80%
One Class SVM [20]	92.5%	79%
RNN [21]	93.7%	69%
DAICS [22]	91.8%	88.9%

5. Conclusion

The dramatic increase of cyber-attacks on industrial control systems can cause huge losses, so identifying potential attacking patterns from ICS network traffic and generating alerts in a timely manner is critically important. In this paper, we propose anomaly detection to detect reconnaissance attacks using isolation forest and deep neural networks by MLP algorithm. The main contribution of this study is examining the efficacy of deep learning to detect reconnaissance attacks and comparing it with the iForest algorithm. The experiments in this study show that a deep neural network performs better having 99% accuracy. In contrast, the iForest tree algorithm has 75% accuracy.

Future Work. Our future work involves implementing other deep learning algorithms, such as Long-Short Term Memory (LSTM) and Convolutional Neural Networks (CNNs), and observing their performance. We also plan to look at other types of attacks, such as command injection and Denial of services (DoS) attacks. We also plan to examine unsupervised deep learning approaches for distinguishing abnormal behavior from normal ones.

Acknowledgement

This work was supported in part by funds from NIST under award number 60NANB18D204, and from NSF under award number CNS 2027750, CNS 1822118 and from NIST, Statnett, Cyber Risk Research, AMI, and ARL.

References

[1] S. East, J. Butts, M. Papa, and S. Shenoi, "A Taxonomy of Attacks on the DNP3 Protocol," in *International Conference on Critical Infrastructure Protection*, 2009, pp. 67–81.

[2] M. A. Teixeira, T. Salman, M. Zolanvari, R. Jain, N. Meskin, and M. Samaka, "SCADA System Testbed for Cybersecurity Research Using Machine Learning Approach," *Future Internet*, vol. 10, no. 8, p. 76, 2018.

[3] P. Huitsing, R. Chandia, M. Papa, and S. Shenoi, "Attack Taxonomies for the Modbus Protocols," *International Journal of Critical Infrastructure Protection*, pp. 37–44, 2008.

[4] K. Stouffer, J. Falco, and K. Scarfone, "Guide to Industrial Control Systems (ICS) Security," *NIST special publication*, vol. 800, no. 82, pp. 16–16, 2011.

[5] A. P. Mathur and N. O. Tippenhauer, "SWaT: A Water Treatment Testbed for Research and Training on ICS Security," in *International Workshop on Cyber-Physical Systems for Smart Sater Networks (CySWater)*, 2016, pp. 31–36.

[6] M. Wan, W. Shang, and P. Zeng, "Double Behavior Characteristics for One-Class Classification Anomaly Detection in Networked Control Systems," *Transactions on Information Forensics and Security*, vol. 12, no. 12, pp. 3011–3023, 2017.

[7] F. Schuster, A. Paul, R. Rietz, and H. König, "Potentials of Using One-Class SVM for Detecting Protocol-Specific Anomalies in Industrial Networks," in *IEEE Symposium Series on Computational Intelligence*, 2015, pp. 83–90.

[8] D. Xu, Y. Wang, Y. Meng, and Z. Zhang, "An Improved Data Anomaly Detection Method Based on Isolation Forest," in *10th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 2, 2017, pp. 287–291.

[9] S. Li, K. Zhang, P. Duan, and X. Kang, "Hyperspectral Anomaly Detection With Kernel Isolation Forest," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 1, pp. 319–329, 2019.

[10] Z. Cheng, C. Zou, and J. Dong, "Outlier Detection Using Isolation Forest and Local Outlier Factor," in *Proceedings of the Conference on Research in Adaptive and Convergent Systems*, 2019, pp. 161–168.

[11] D. Myers, S. Suriadi, K. Radke, and E. Foo, "Anomaly Detection for Industrial Control Systems Using Process Mining," *Computers & Security*, vol. 78, pp. 103–125, 2018.

[12] P.-H. Wang, I.-E. Liao, K.-F. Kao, and J.-Y. Huang, "An Intrusion Detection Method Based on Log Sequence Clustering of Honeypot for Modbus TCP Protocol," in *International Conference on Applied System Invention (ICASI)*, 2018, pp. 255–258.

[13] C. Wang, B. Wang, H. Liu, and H. Qu, "Anomaly Detection for Industrial Control System Based on Autoencoder Neural Network," *Wireless Communications and Mobile Computing*, vol. 2020, 2020.

[14] S. Kim, W. Jo, and T. Shon, "APAD: Autoencoder-Based Payload Anomaly Detection for Industrial IoE," *Applied Soft Computing*, vol. 88, p. 106017, 2020.

[15] Á. L. P. Gómez, L. F. Maimó, A. H. Celdran, F. J. G. Clemente, C. C. Sarmiento, C. J. D. C. Masa, and R. M. Nistal, "On the Generation of Anomaly Detection Datasets in Industrial Control Systems," *IEEE Access*, vol. 7, pp. 177 460–177 473, 2019.

[16] J. Goh, S. Adepu, K. N. Junejo, and A. Mathur, "A Dataset to Support Research in the Design of Secure Water Treatment Systems," in *International Conference on Critical Information Infrastructures Security*. Springer, 2016, pp. 88–99.

[17] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation

Forest," in *2008 Eighth International Conference on data Mining*, 2008, pp. 413–422.

[18] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API Design for Machine Learning Software: Experiences From the Scikit-Learn Project," in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Database (ECML PKDD) Workshop: Languages for Data Mining and Machine Learning*, 2013, pp. 108–122.

[19] H. Taud and J. Mas, "Multilayer Perceptron (MLP)," in *Geomatic Approaches for Modeling Land Change Scenarios*, 2018, pp. 451–455.

[20] J. Inoue, Y. Yamagata, Y. Chen, C. M. Poskitt, and J. Sun, "Anomaly Detection for a Water Treatment System Using Unsupervised Machine Learning," in *IEEE International Conference on Data Mining Workshops (ICDMW)*, 2017, pp. 1058–1065.

[21] D. Shalyga, P. Filonov, and A. Lavrentyev, "Anomaly Detection for Water Treatment System based on Neural Network With Automatic Architecture Optimization," *ICML Workshop for Deep Learning for Safety-Critical in Engineering Systems*, 2018.

[22] M. F. Abdelaty, R. D. Corin, and D. Siracusa, "DAICS: A Deep Learning Solution for Anomaly Detection in Industrial Control Systems," *IEEE Transactions on Emerging Topics in Computing*, 2021.