A Dynamic Network-based Intrusion Detection Model for Industrial Control Systems

Paulo R. de Oliveira*, Altair O. Santin*, Pedro Horchulhack*, Eduardo K. Viegas*, Everton de Matos†
*Pontifical Catholic University of Paraná, Curitiba, Paraná, 80215-901, Brazil.
{paulo.oliveira, santin, pedro.horchulhack, eduardo.viegas}@ppgia.pucpr.br
†Technology Innovation Institute, Abu Dhabi - United Arab Emirates.
everton.dematos@tii.ae

***Abstract*—Industrial Control Systems (ICS) play a crucial role in managing and controlling industrial assets. Due to their critical importance, adversaries are often highly motivated to target these systems, as a successful attack can disrupt the entire industry's operations. In general, to improve the system's security, proposed intrusion detection schemes often resort to traditional security mechanisms. As a consequence, due to their static nature, attackers can easily evade designed detection approaches. In light of this, this paper proposes a new dynamic network-based intrusion detection model for ICS, implemented in two phases. First, our scheme extracts network-related features to describe the current ICS environment behavior. Second, the security mechanisms are proactively selected based on the extracted network traffic behavior. As a result, our scheme can adjust the system's configuration based on the current assessed event. Experiments on a new dataset, featuring over 14 attack categories targeting a SCADA system revealed that traditional detection methods face challenges in handling diverse attack categories. Conversely, our proposed model improved the average true-positive rates by up to 20% while also improving the range of detected attacks.**

***Index Terms*—SCADA, Industrial Control Systems, Machine Learning**

I. Introduction

In the past few years, *behavior-based* Network-based Intrusion Detection System (NIDS) employing Machine Learning (ML) strategies have shown remarkable accuracy [1]. These systems construct ML models by observing the training dataset behavior while assessing them using testing datasets that mimic real-world scenarios [2]. However, even with the encouraging outcomes, the application of ML in NIDS for Industrial Control System (ICS) predominantly remains within the research domain [3].

ICS combine hardware, software, and network systems to improve and automate operations in various industries, including manufacturing, energy, and transportation [4]. Supervisory Control and Data Acquisition (SCADA) systems play a critical role within ICS, as they automate the control of Programmable Logic Controller (PLC) through Human-Machine Interface (HMI). To achieve such a goal, SCADA use multiple networking protocols such as Modbus, Distributed Network Protocol 3 (DNP3), and Open Platform Communications (OPC) to ensure smooth communication and data transfer [5].

Driven by ICS systems' essential and sensitive role, adversaries meticulously study the target infrastructure for a long duration to find and leverage multiple undiscovered vulnerabilities for their disruption [6]. In the past few years, security breaches have led to failures in the power grid, resulting in disruptions and potential economic and societal losses [7]. As a result, to thoroughly protect these systems, it is crucial to use multiple security mechanisms, such as authentication, authorization, firewalls, and VPNs [8]. In such a context, to examine the network traffic within an ICS, operators often use NIDS tools.

Many strategies use either *misuse-based* or *behavior-based* methods [9] for effective intrusion detection. Misuse-based methods rely on recognized attack patterns, which protect against known intrusions but may fail to protect against emerging threats. On the other hand, *behavior-based* strategies focus on examining the events behavior for any deviation from a predefined baseline, thus demonstrating their detection potential for unknown attacks.

ICS setups, including SCADA, are significant magnets for persistent attackers who carefully analyze their targets to execute successful attacks [10]. As a result, static ML-based intrusion detection systems face evasion risks as their model parameters can only be changed after training. The emergence of novel attack vectors over time increases the probability of evading the established system, even without clear insight into the model behavior by attackers. As a result, existing ML-oriented NIDSs need to actively select and apply security strategies to strengthen the system's security.

Contribution. In this paper, we present an innovative NIDS approach for ICS that aims to improve classification accuracy by dynamically adapting the model configuration to the current system inputs. A dynamic classifier selection method is proposed to perform the classification. This method dynamically selects the optimal classifiers based on the prevailing event behavior. Multiple security mechanisms are integrated with our proposed model. In synchronization with the ongoing event behavior, it actively determines the appropriate tools for the detection task. This strategy increases the classification accuracy by dynamically modifying the system setup based on the analyzed event attributes. To conclude, our proposed scheme can improve the accuracy of the system while increasing the system's resilience against sophisticated types of attacks.

Roadmap. The organization of the rest of this document is as outlined below. Section II provides the background on

ICS and Intrusion Detection System (IDS) methodologies. The Section III summarizes the related work in ICS intrusion detection. The Dynamic NIDS model for ICS is presented in Section IV, followed by the evaluation outcomes in Section V-B. The paper is drawn to a close in Section VI.

II. Background

In this section, we offer an extensive overview of ICS, including SCADA, and its principal associated infrastructure components. Additionally, we introduce the fundamental components of ML commonly employed in NIDS.

A. Industrial Control Systems

An ICS is a complex, typically legacy system essential for managing hardware, software, and network elements in various critical infrastructure sectors, including manufacturing, oil and gas, energy, and water treatment [4]. To fulfill this purpose, a typical ICS comprises three primary components: SCADA, PLC, and communication elements. SCADA oversees and controls industrial assets through an HMI, enabling data management, alerts, and control commands. PLC are digital devices that control industrial machinery and processes, acting as an interface between the SCADA system or other controllers to facilitate interaction with sensors and actuators. Finally, the communication components use diverse protocols like Ethernet, Profibus, and Modbus to enable interaction between the SCADA system and the PLC.

ICSs, as cyber-connected systems, are exposed to motivated attackers seeking to disrupt critical industrial infrastructure, potentially leading to control over PLCs and resulting in significant losses [11]. As an example, in 2015, the BlackEnergy malware infected Ukrainian SCADA systems within a power grid ICS, causing extensive power outages that impacted more than 230 thousand consumers [12]. Securing ICSs is challenging due to their interconnected and difficult-to-update architecture, necessitating multiple security measures such as authentication, authorization, firewalls, VPNs, and NIDS for reinforcement [13].

B. Network-based Intrusion Detection System

Behavior-based NIDSs typically employ ML-based pattern recognition to monitor and analyze network activities [14]. The process is typically implemented by making use of four main modules:

- *Data Acquisition*. Collects network events continuously, such as gathering network packets from a Network Interface Card (NIC).
- *Feature Extraction*. Derives behavior from collected data, representing it as network flows that summarize the interactions between network entities within a specified time window.
- *Classification*. Utilizes a ML model to classify the input as either *normal* or *intrusion* using the extracted feature set.
- *Alert*. Signal events classified as intrusions to the network operator.

In practice, the reliability of behavior-based NIDSs depends on the ML model's quality [15]. However, the creation of realistic intrusion datasets for ICS systems is a challenging task [16], as attackers often use zero-day attacks to compromise these critical systems. Conversely, existing ML-based NIDSs for ICSs often rely on traditional pattern recognition, which may not provide the required level of detection reliability [17].

III. Related Works

To improve the ICS security, researchers generally concentrate on enhancing detection accuracy through ML-based methods [18]. Louk *et al.* [19] employs an ensemble of classifiers to enhance the detection of new attacks when evaluated on a power grid dataset. Another ensemble approach was proposed by R. Lazzarini *et al.* [20] where the authors stack several Deep Neural Network (DNN) models to improve detection accuracy in a multi-class dataset. Their proposed scheme reached low false-positive rates on a binary classification dataset encompassing Internet of Things (IoT) related attacks. Similarly, J. Gao *et al.* [21] proposed the usage of DNN to detect SCADA related attacks. The authors showed that their proposed model was able to provide significantly high detection accuracies while detecting temporally uncorrelated attacks.

In general, detection reliability improvements are achieved through the combination of multiple classification systems. As an example, V. Ravi *et al.* [22] relies on a feature fusion through a Recurrent Neural Network (RNN) to improve detection accuracy on a network intrusion dataset. Their proposed model was able to significantly improve detection accuracy, however, they overlooked detection of new kinds of attacks. A similar approach was proposed by Y. Li *et al.* [23] where the authors relies on a two-stage training to improve detection of an ensemble of classifiers. The authors improved the detection accuracy on a ICS-related dataset. However, detection of new attacks was not addressed and the authors assumed that similar accuracies would be observed when subject to unknown threats. M. S. Al-Daweri *et al.* [24] proposed the application of feature selection to build one-class DNN models for intrusion detection in ICS. The proposed model significantly improved detection accuracy on multiple datasets while not considering the detection of new attacks.

The detection of new kinds of attacks in ICS has been the subject of multiple works in the literature over the past few years [25]. M. Catillo *et al.* [26] proposed applying a semi-supervised technique through a deep autoencoder to detect intrusions as outliers. The authors proposed model could detect attacks not used during the training phase in ICS-related dataset. I. A. Khan *et al.* [27] proposed the application of anomaly detection techniques to detect new SCADA attacks. Their proposed model was able to detect new kinds of attacks while also improving the detection accuracy of known intrusions. T. D. Ramotsoela *et al.* [28] proposed a similar approach using DNN models, showing their capability to detect new attacks. However, although proposed solutions can detect new kinds of attacks, they leave systems unprotected against highly motivated attackers. In such a case, the adversary can still maliciously craft the generated attack behavior to evade deployed security solutions.

IV. Dynamic NIDS Model

To address such a shortcoming, we present a new reliable intrusion detection model that addresses the detection of advanced attacks on ICS systems. Our design model considers a Operational Technology (OT) network traffic environment where multiple security mechanisms are used to monitor the network traffic that passes through. In particular, we monitor the network traffic received by a SCADA system. Our proposed scheme is shown in Figure 1 and is implemented in two main phases, namely *Security Mechanisms Pool* and *Combination*.

Our solution's security mechanism is implemented by the *Security Mechanism Pool*. A data collection module continuously collects the passing network traffic and provides the collected data to the deployed security mechanisms. Each considered security mechanism evaluates the current network traffic behavior and generates a network traffic decision as *normal* or *intrusion*. Our model considers that multiple security solutions can be used to achieve detection, such as firewalls, *behavior-based* NIDS and *signature-based* NIDS. The generated decisions are fed as input to the *Combination* module.

The *Combination* module goal is to evaluate the decisions generated by each security mechanism to reach a final decision. To achieve such a goal, we model the combination as a dynamic classification selection task, dynamically adjusting the tools for detection based on the current network traffic behavior. As a result, our scheme can increase the system accuracy even in the presence of new categories of network attacks.

In the following subsections, we further describe our proposed model, including how to implement it.

A. Security Mechanism Pool

Current techniques used in the literature often use a single security mechanism to enhance the ICS security. For example, authors often rely on a single *behavior-based* NIDS configuration to detect intrusion attempts, exposing their system to highly motivated attackers who may evade the deployed mechanism.

To address this shortcoming, our proposed model incorporates multiple security mechanisms. For example, we consider a deployment with a firewall, a *behavior-based* NIDS, and a *signature-based* NIDS (*Security Mechanism Pool*, Fig. 1). As time progresses, each deployed security solution assesses the OT network traffic. Each scheme generates a decision, which is then utilized as input by our *Combination* module. This module aims to proactively determine the most effective set of tools for detecting the generated attack.

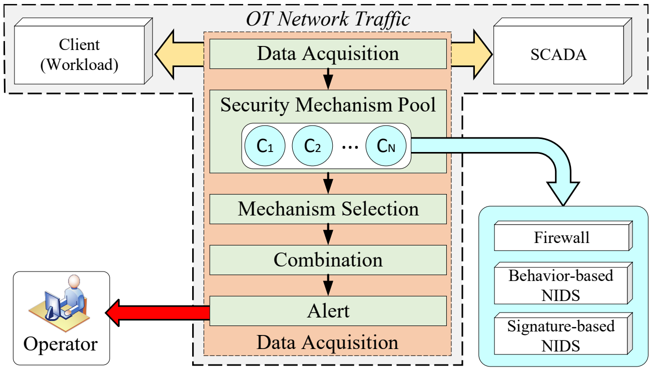

Fig. 1: A reliable intrusion detection model for industrial control systems.

B. Combination

The combination module's objective is to evaluate the predictions made by deployed security mechanisms. In practical terms, it selects the security mechanisms based on current network traffic behavior. To accomplish this, we extract and assess network traffic behavior using a dataset of previously correctly classified events.

For each event in a given training dataset $\mathcal{D}$, the module extracts a set of network flow features to create a feature vector x. This resulting vector, along with the prediction from each security mechanism SM_i, is evaluated. If SM_i correctly classifies the event, it is recorded in a reliable dataset named $\mathcal{D}_{correct}$. The dataset is used as a selection criterion for newly evaluated events.

At deployment time, the behavior of the collected event is extracted to compound a feature vector x. The module then proceeds to find the top k events on the $\mathcal{D}_{correct}$ closest to feature vector x, through the following equation:

$$\underset{x_i \in \mathcal{D}_{correct}}{\text{argtop k}} \sqrt{\sum_{j=1}^{N}(x^j - x_i^j)^2} \tag{1}$$

where k represents the number of neighboring events, and N signifies the number of features. Consequently, our scheme identifies the k events closest to the assessed event x through an Euclidean distance. These identified events are aggregated using a majority voting procedure to determine the event class. Hence, our model can dynamically adapt the selection of security mechanisms for event detection based on the current network traffic behavior, thereby enhancing system accuracy through this adjustment.

V. Evaluation

This section evaluates the proposed reliable network-based intrusion detection model in ICS. To achieve such a goal, the following subsections outline the construction of the used models in the evaluation and describe their performance in the testbed.

TABLE I: Features set extracted at the network level for each feature grouping in a time window interval of 15s.

Grouping	Description
Source / Destination	Number of Packets
	Number of Bytes
	Minimum packet length
	Maximum packet length
	Average packet length
	Packet length standard deviation
	Minimum inter-arrival-time
	Maximum inter-arrival-time
	Average inter-arrival-time
	Total Packet length
Flow	Number of packets
	Number of Bytes
	Number of pushed packets
	Active time
	Maximum active time
	Minimum active time

TABLE II: Detection accuracy for each network traffic on our testbed according to the detection approach.

#	Behavior	***Detection Accuracy (%)***			
		Traditional			**Our Approach**
		Firewall	**Behavior NIDS**	**Misuse NIDS**	
1	XSS	40.5	50	90.9	48.0
2	Code Injection	50	95.6	95.6	99.0
3	Read Reg.	96.6	99.7	50	100
4	DOS Write Reg.	72.5	97.2	50	100
5	DOS Write Coils	88.2	50	50	100
6	Portscan	50	50	100	100
7	Write Single Coils	88.2	50	50	100
8	Advanced scan	44.9	50	98	99.9
9	Read Input Reg.	96.6	99.7	50	100
10	Scanner UID	44.3	50	50	100
11	SQL Injection	50	50	81.9	50.9
12	Read Coils	95.3	99.5	50	100
13	Scanner	52.6	50	100	50
14	Write Reg.	91.6	99.7	50	100

A. *Prototype*

We have deployed our proposal prototype in a distributed environment. This setup includes a SCADA system using ScadaBR $v.1.0CE$. The incoming OT network traffic can be evaluated by three security tools, as follows: (*i*) firewall configured through Sahu *et al.* [29] rules; (*ii*) *behavior-based* NIDS implemented using various classifiers (see Section V-B) based on the network flow features extracted using the FlowTBag feature extractor [30] as listed in Table I; and a (*iii*) *misuse-based* NIDS implemented through Snort with the *snort3* community rules.

To simulate diverse OT-related network traffic, we used well-known workload tools for Modbus, DNP3, OPC, and web service protocols. Our *Data Acquisition* module, based on SCAPY $v.2.5.0$, collected this traffic. Network features were extracted using the FlowTBag feature extractor [30] in 15-second intervals. The *Combination* module (Eq. 1) was implemented with the scikit-learn API using the Euclidean distance API.

We deployed the prototype in a controlled testbed for data collection, involving 100 client workload machines for generating normal network traffic. We injected attacker-related network traffic at irregular intervals towards the deployed SCADA, utilizing 14 attacker machines for the generation of each specific attack. This testbed operated for a total of 96 hours. During the testbed execution, we recorded outputs from deployed security mechanisms.

We evaluate the performance of our model using the deployed testbed. To achieve this goal, we consider a scenario where new attacks are continually generated by creating three datasets as follows: (*Training*) A dataset encompassing a total of 5 attack categories and 40% of normal traffic; (*Validation*) A dataset encompassing a total of 6 attack categories and 30% of normal traffic; (*Testing*) A dataset encompassing a total of 11 attack categories and 30% of normal traffic;

The datasets were generated using the FlowTBag feature extractor [30], summarizing communication in 15-second intervals. To address dataset imbalance, we applied random undersampling without replacement. The data was then normalized using minimum-maximum range scaling to values between 0 and 1.

The selected techniques were evaluated with respect to their True-Positive (TP), True-Negative (TN), and F1 scores by considering the following classification performance metrics:

- *True-Positive* (TP): rate of *intrusion* samples correctly classified as *intrusion*.
- *True-Positive* (TN): rate of *normal* samples correctly classified as *normal*.
- *False-Positive* (FP): rate of *normal* samples incorrectly classified as *intrusion*.
- *False-Negative* (FN): rate of *intrusion* samples incorrectly classified as *normal*.

B. *ICS Intrusion Detection*

Our first experiment aims to investigate the detection performance of the selected security mechanisms (Fig. 1, *Security Mechanism Pool*). More specifically, we assess the detection accuracy of the deployed security tools (described in Section V-A) while detecting the generated 14 attack behaviors. To accomplish this objective, the generated set of attacks were evaluated by three security mechanisms as follows: (*i*) Firewall configured using rules from Sahu *et al.* [29]; (*ii*) Behavior-based NIDS employing a naive bayes classifier, with network flow features extracted using the FlowTBag feature extractor [30] (refer to Table I); (*iii*) Misuse-based NIDS implemented with Snort utilizing the *snort3* community rules. The evaluation aims to thoroughly assess the performance of the deployed security solutions in countering the generated set of attacks.

Table II shows the detection accuracy of the deployed security solutions concerning each generated attack (*Behavior*) and the used security mechanism (*Traditional*). Generally, the employed security solutions face challenges in achieving high

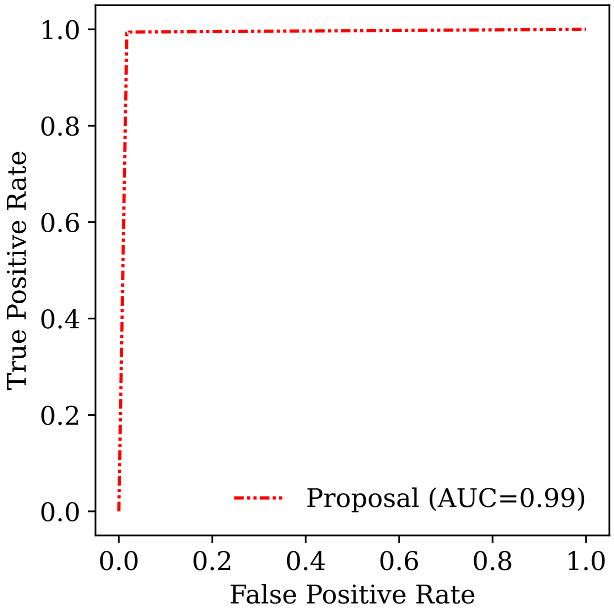

Fig. 2: Detection performance of our proposed dynamic selection model with and without the classification assessment.

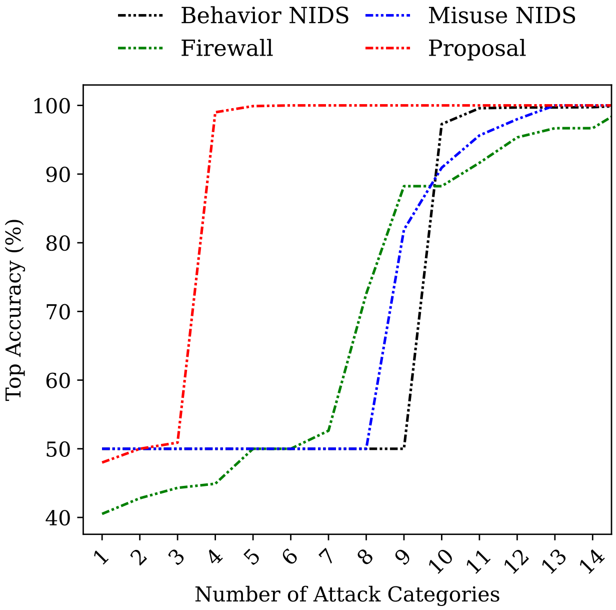

Fig. 3: Sorted top detection accuracy for each selected intrusion detection technique.

detection accuracy across multiple attacks. For instance, the Firewall-based detection achieved TP rates exceeding 90% for only 4 out of 14 generated attacks. Similarly, the misuse-based and behavior-based NIDS respectively detected only 6 and 4 attacks with TP rates exceeding 90%. On average, the TP rates for the used security solutions were approximately 66.7%, 70.8%, and 69.0% for Firewall, Behavior-based, and Misuse-based NIDS, respectively. The results imply that attackers can effectively evade the deployed security solutions by altering their attack behavior. This is evident as most of the generated attacks, which were not part of the behavior-based NIDS training data and lack related Firewall rules or misuse-based NIDS signatures, evade detection, leading to low TP rates. This situation is exacerbated in ICS scenarios where attackers are highly motivated to target critical systems. Current security mechanisms fall short on achieving the required level of detection reliability for ICS environments.

Our second experiment aims to assess how our proposed model can enhance ICS security. We implement the combination module (Eq.1) to merge decisions from multiple security mechanisms. During the experiments, we set k as 10 and performed Euclidean distance computation using scikit-learn. For each network event, we extract network flow features (Table I) and apply our combination criteria to determine the appropriate security solutions that should be used. Using neighboring events, we employ a majority voting procedure to trigger alerts for the network operator. The parameters were empirically configured, with no significant variations observed during testing

Figure 2 shows the Receiver Operating Characteristic (ROC) curve of our proposed model on the evaluated dataset. It is possible to observe that we achieved significantly high detection accuracies, reaching an Area Under the Curve (AUC) value of 0.99. We further investigate how our proposed model performs for the detection of different categories of attacks on our deployed testbed.

Table II shows the accuracy performance of our model for each evaluates attack type under the *Our Approach* listing. Our proposed scheme significantly improve detection accuracy when compared to traditional single-based security mechanism techniques. As an example, considering a 90% TP rate threshold, our proposed model is able to detect 11 out of the 14 attacks. This result in a significant improvement when compared to traditional approaches, which can only reach the same level of reliability for 6 attack categories in the best-case scenario (Table II, *Misuse-based* NIDS). In practice, our proposed scheme was able to reach an average detection accuracy of 89.1%, presenting an improvement of up to 20% when compared to the traditional approaches (Table II, Firewall). As a result, our proposed model is able to improve the reliability of intrusion detection in ICS deployments.

We further explore the accuracy advantages of our model in Figure 3. Our scheme achieves a minimum of 80% detection accuracy for 13 out of the 14 evaluated attacks. In contrast, traditional schemes achieve 80% accuracy for only approximately 5 out of the 14 attacks. This highlights substantial enhancements in detection reliability, promising more dependable intrusion detection in ICS deployments. The benefits brought by our solution is thanks to the application of the proactive selection of the used security solution based on the current network traffic behavior.

VI. Conclusion

In the literature, intrusion detection by means of a single security strategy leaves the systems unprotected against highly motivated attackers targeting ICS.

This paper proposes a new dynamic intrusion detection model that includes a combination strategy to combine multiple security solutions. Our proposed model evaluates the current network traffic behavior and proactively selects the security mechanisms that should be used for effective and reliable intrusion detection. Experiments that have been conducted on a novel dataset that includes 14 categories of attacks have demonstrated the feasibility of our proposed model.

As future works, we plan on extending our solution to evaluate the reliability of the decision. This includes alerting the operator in the case that none of the security solutions in place can classify the event being evaluated with a high level of confidence.

Acknowledgment

This work was partially sponsored by the Brazilian National Council for Scientific and Technological Development (CNPq) grant n. 304990/2021-3.

References

[1] E. K. Viegas, A. O. Santin, and P. Tedeschi, "Toward a reliable evaluation of machine learning schemes for network-based intrusion detection," *IEEE Internet of Things Magazine*, vol. 6, no. 2, pp. 70–75, Jun. 2023.

[2] B. B. Bulle, A. O. Santin, E. K. Viegas, and R. R. dos Santos, "A host-based intrusion detection model based on OS diversity for SCADA," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, Oct. 2020.

[3] D. Arp, E. Quiring, F. Pendlebury, A. Warnecke, F. Pierazzi, C. Wressnegger, L. Cavallaro, and K. Rieck, "Dos and don'ts of machine learning in computer security," in *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Aug. 2022, pp. 3971–3988.

[4] H. Kayan, M. Nunes, O. Rana, P. Burnap, and C. Perera, "Cybersecurity of industrial cyber-physical systems: A review," *ACM Comput. Surv.*, vol. 54, no. 11s, sep 2022.

[5] M. Alanazi, A. Mahmood, and M. J. M. Chowdhury, "Scada vulnerabilities and attacks: A review of the state-of-the-art and open issues," *Computers Security*, vol. 125, p. 103028, 2023.

[6] C. Sheng, Y. Yao, W. Li, W. Yang, and Y. Liu, "Unknown attack traffic classification in scada network using heuristic clustering technique," *IEEE Transactions on Network and Service Management*, pp. 1–1, 2023.

[7] W. Alsabbagh, S. Amogbonjaye, D. Urrego, and P. Langendörfer, "A stealthy false command injection attack on modbus based scada systems," in *2023 IEEE 20th Consumer Communications Networking Conference (CCNC)*, 2023, pp. 1–9.

[8] V. Abreu, A. O. Santin, E. K. Viegas, and M. Stihler, "A multi-domain role activation model," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, May 2017. [Online]. Available: https://doi.org/10.1109/icc.2017.7997247

[9] R. R. dos Santos, E. K. Viegas, A. O. Santin, and V. V. Cogo, "Reinforcement learning for intrusion detection: More model longness and fewer updates," *IEEE Transactions on Network and Service Management*, vol. 20, no. 2, pp. 2040–2055, Jun. 2023.

[10] J. Qian, X. Du, B. Chen, B. Qu, K. Zeng, and J. Liu, "Cyber-physical integrated intrusion detection scheme in scada system of process manufacturing industry," *IEEE Access*, vol. 8, pp. 147 471–147 481, 2020.

[11] A. T. A. Ghazo and R. Kumar, "Critical attacks set identification in attack graphs for computer and scada/ics networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–0, 2023.

[12] M. Geiger, J. Bauer, M. Masuch, and J. Franke, "An analysis of black energy 3, crashoverride, and trisis, three malware approaches targeting operational technology systems," in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1, 2020, pp. 1537–1543.

[13] T. Sasaki, A. Fujita, C. H. Ganán, M. van Eeten, K. Yoshioka, and T. Matsumoto, "Exposed infrastructures: Discovery, attacks and remediation of insecure ics remote management devices," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 2379–2396.

[14] E. Viegas, A. Santin, V. Abreu, and L. S. Oliveira, "Enabling anomaly-based intrusion detection through model generalization," in *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, Jun. 2018.

[15] J. Geremias, E. K. Viegas, A. O. Santin, A. Britto, and P. Horchulhack, "Towards multi-view android malware detection through image-based deep learning," in *2022 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, May 2022.

[16] R. R. dos Santos, E. K. Viegas, A. O. Santin, and P. Tedeschi, "Federated learning for reliable model updates in network-based intrusion detection," *Computers &amp Security*, vol. 133, p. 103413, Oct. 2023. [Online]. Available: https://doi.org/10.1016/j.cose.2023.103413

[17] S. Choi, J. Choi, J.-H. Yun, B.-G. Min, and H. Kim, "Expansion of ICS testbed for security validation based on MITRE ATT&CK techniques," in *13th USENIX Workshop on Cyber Security Experimentation and Test (CSET 20)*. USENIX Association, Aug. 2020. [Online]. Available: https://www.usenix.org/conference/cset20/presentation/choi

[18] D. Bhamare, M. Zolanvari, A. Erbad, R. Jain, K. Khan, and N. Meskin, "Cybersecurity for industrial control systems: A survey," *Computers &amp Security*, vol. 89, p. 101677, Feb. 2020. [Online]. Available: https://doi.org/10.1016/j.cose.2019.101677

[19] M. H. L. Louk and B. A. Tama, "Exploring ensemble-based class imbalance learners for intrusion detection in industrial control networks," *Big Data and Cognitive Computing*, vol. 5, no. 4, p. 72, Dec. 2021. [Online]. Available: https://doi.org/10.3390/bdcc5040072

[20] R. Lazzarini, H. Tianfield, and V. Charissis, "A stacking ensemble of deep learning models for IoT intrusion detection," *Knowledge-Based Systems*, vol. 279, p. 110941, Nov. 2023. [Online]. Available: https://doi.org/10.1016/j.knosys.2023.110941

[21] J. Gao, L. Gan, F. Buschendorf, L. Zhang, H. Liu, P. Li, X. Dong, and T. Lu, "Omni SCADA intrusion detection using deep learning algorithms," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 951–961, Jan. 2021. [Online]. Available: https://doi.org/10.1109/jiot.2020.3009180

[22] V. Ravi, R. Chaganti, and M. Alazab, "Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system," *Computers and Electrical Engineering*, vol. 102, p. 108156, Sep. 2022.

[23] Y. Li, W. Xue, T. Wu, H. Wang, B. Zhou, S. Aziz, and Y. He, "Intrusion detection of cyber physical energy system based on multivariate ensemble classification," *Energy*, vol. 218, p. 119505, Mar. 2021. [Online]. Available: https://doi.org/10.1016/j.energy.2020.119505

[24] M. S. Al-Daweri, S. Abdullah, and K. A. Z. Ariffin, "A homogeneous ensemble based dynamic artificial neural network for solving the intrusion detection problem," *International Journal of Critical Infrastructure Protection*, vol. 34, p. 100449, Sep. 2021. [Online]. Available: https://doi.org/10.1016/j.ijcip.2021.100449

[25] K. Bansal and A. Singhrova, "Review on intrusion detection system for IoT/IIoT -brief study," *Multimedia Tools and Applications*, Aug. 2023. [Online]. Available: https://doi.org/10.1007/s11042-023-16395-6

[26] M. Catillo, A. Pecchia, and U. Villano, "CPS-GUARD: Intrusion detection for cyber-physical systems and IoT devices using outlier-aware deep autoencoders," *Computers &amp Security*, vol. 129, p. 103210, Jun. 2023. [Online]. Available: https://doi.org/10.1016/j.cose.2023.103210

[27] I. A. Khan, D. Pi, Z. U. Khan, Y. Hussain, and A. Nawaz, "HML-IDS: A hybrid-multilevel anomaly prediction approach for intrusion detection in SCADA systems," *IEEE Access*, vol. 7, pp. 89 507–89 521, 2019. [Online]. Available: https://doi.org/10.1109/access.2019.2925838

[28] T. D. Ramotsoela, G. P. Hancke, and A. M. Abu-Mahfouz, "Behavioural intrusion detection in water distribution systems using neural networks," *IEEE Access*, vol. 8, pp. 190 403–190 416, 2020.

[29] A. Sahu, P. Wlazlo, N. Gaudet, A. Goulart, E. Rogers, and K. Davis, "Generation of firewall configurations for a large scale synthetic power system," in *2022 IEEE Texas Power and Energy Conference (TPEC)*, 2022, pp. 1–6.

[30] L. C. B. Guimarães, G. A. F. Rebello, G. F. Camilo, L. A. C. de Souza, and O. C. M. B. Duarte, "A threat monitoring system for intelligent data analytics of network traffic," *Annals of Telecommunications*, vol. 77, no. 7-8, pp. 539–554, Oct. 2021. [Online]. Available: https://doi.org/10.1007/s12243-021-00893-5