Attack Learning Rate: An Investigation on Intrusion Detection for SCADA

Devon Kelly
Virginia Tech
Blacksburg, VA
devonkel4@vt.edu

James Agresto
Virginia Tech
Blacksburg, VA
agrestjj@vt.edu

ABSTRACT

This project focused on analyzing the learning rate for machine learning models for anomaly detection within an operational technology network, particularly one with Supervisory Control and Data Acquisition (SCADA) systems. For this project, an amplification was made to traditional intrusion detection datasets to further expand its breadth and test to ensure models would be equally receptive towards the generated attack as with the ones included in the datasets. On top of this the models were compared against a dataset geared for a SCADA environment and the attacks inherent to the landscape. Whereas other papers focus on the accuracy metrics alone, this project sought to analyze the learning rate of models and improvement in accuracy as the amount of data became available, emphasizing early accuracy above overall accuracy. Through testing, it was found that Random Forest was the quickest-learning lightweight model for this project as it achieved above 95% accuracy with around 8 data points consistently across many seeds. With the lightweight deployment that is possible with Random Forest, we envision feasible deployment in a real world SCADA environment since it can run on robust hardware as opposed to models most closely associated with deployment within cloud architecture due to on-site hardware limitations. Furthermore, with a human in the loop, the model can be asymmetrically updated and redeployed without interrupting communications or putting strain on the intrusion detection system installed in the network.

1 INTRODUCTION

Intrusion Detection Systems (IDS) are crucial for preventing attacks in many different network environments. With the popularization of artificial intelligence and machine learning in the research landscape, many approaches at creating machine learning-based IDS mechanisms. One gap in the current landscape is the lack of investigation towards how quickly the best models can learn, instead focusing on the final results. The models are trained for general network environments, where the needs are balanced in such a way that confidentiality, integrity, and accessibility are near equally focused.

For more specialized network environments such as Operational Technology (OT) networks, more specialized datasets and anomalies must be supervised. OT networks host not only the most critical tasks, but also the most vulnerabilities. OT networks handle wide ranges of tasks from heavy machinery, HVAC systems, and other important industrial tasks. These services come with risk to human life and expensive costs for down times in environments such as factories. Not only that, but these networks host embedded systems and controllers on legacy hardware, since the devices are built to be robust and last for longer lifespans than traditional Information Technology (IT) network devices. It is generally expensive to replace these devices due to highly specific energy and timing requirements, as well as compatibility with the other devices. The major downside to this is that these devices are also vulnerable to many more attacks than traditional IT infrastructure.

1.1 IT and OT Communication

1.1.1 OT Network Requirements. Machines within an OT network are highly delay sensitive. The PLCs and embedded systems deployed typically run either real-time operating systems, older operating systems in stripped down formats, or very light weight systems to minimize latency. Along side this these systems use very little encryption or cybersecurity systems due to the processing demand many of these techniques require. Because of this, if an attacker was able to gain access to the OT network, it would almost be guaranteed that the devices present would become completely compromised. The PLCs and systems present on the OT network are the backbone of many high-risk and high-dollar operations. As such, any change in behavior, especially from purposefully malicious actions, can jeopardize human life and disable operations until recovery is made, potentially costing millions of dollars even for short downtime.

1.1.2 One-Way Traffic and Air-Gapping. With the sensitive nature of the OT network as a whole, many organizations have previously made it a priority to seclude the OT network from the remainder of infrastructure. One of these techniques used is called air-gapping, the practice of protecting an area by preventing any network connection between the sensitive area and the rest of the network. This technique prevents all forms of exploitation except ones stemming from physical security weaknesses since there is no connection to other areas of the network to leverage from. The main downside of this approach however is that it limits monitoring and analysis of devices and data on the air-gapped network to on-site operators and machines on the right side of the air-gapping.

Many organizations require the use of data from these networks, and wish for access further than on-site only operation. The way around it is to implement some form of one-way traffic sourcing from the OT network out to the IT network, disallowing traffic back inwards. This is accomplishable through various means including data diodes, as well as firewall rules, routing rules, among others. This technique sacrifices the confidentiality of the exported data to not only gain access to read-only data from the OT network, but also assists in limiting the amount of people who require on-site access to the OT network. Since if the data is accessible on the less-sensitive IT network, only operators and contractors are required access to the OT network, while analysts and people working with the data can get what they need from the IT side.

With the current shift towards bidirectional communication, commonly used is the usage of DMZ firewalls to control access between the IT and OT networks. It is a strict firewall that typically allows outbound packets from the OT network into the IT network, while blocking the vast majority of destinations and ports and maintaining a strict whitelist of allowed hosts capable of communicating with the few open destinations. This format does begin to allow for an attack surface into the OT network, since if a whitelisted host gains access into the network, they can infiltrate the OT network more directly using the compromised host as leverage.

1.1.3 New Demands. Recently remote data acquisition is not where demands have stopped. The new standard is incorporating the idea of remote management and cross-network communication with other OT networks. Especially in the context of the growth of work-from-home in corporate environments following COVID-19 related health issues and lockdowns worldwide, interest in remote operation has significantly grown. This new demand actively subverts the inherent protections provided by the previous techniques that prevent inbound traffic into the OT network. The benefit however is that in the course of normal operation, a given controller's operation could be adjusted over VPN connection from home, different locations can synchronize their operations based on demand through Industrial Internet of Things (IIoT) operation, and AI can be connected to optimize certain outputs, among many other use cases. Not only that, but IIoT operations bypass the DMZ firewalls deployed in the process of their communication.

1.2 What is IIoT

The Industrial Internet of Things, also known as IIoT is a recent development used to optimize operational technology setups. IIoT devices increase connectivity and data transmission between OT devices, and facilities as well. The development of these devices stemmed from a desire to use aggregated data to create predictions, change models, and allow automatic adjustments in automation-driven industrial environments. The birth of this technology, combined with other recent technologies, such as advances in AI, edge computing, and cloud computing, has driven what is commonly agreed to be the fourth industrial revolution or industry 4.0 [1]. The key feature of Industry 4.0 is the ability for automated environments such as manufacturing facilities to seamlessly incorporate adjustments based on data in order to reduce human oversight.

NISTIR 8259 [2] defines an Internet of Things (IoT) device as a device that has "at least one transducer for interacting directly with the physical world and at least one network interface for interfacing with the digital world." This definition has been used by US law and was introduced by the National Institute of Standards and Technology (NIST).

1.2.1 IoT vs IIoT. IoT devices, also commonly known as smart devices, have been improving homes and many industries as a whole. These devices perform a user-facing function and use increased connectivity to provide more accessibility to the end user. These devices include smart thermostats, smart home hubs, and home automation kits. IoT systems also take users from manually controlling a device to using some adjusted automated process to perform the action for them. For example, a regular A/C control requires human input to determine the set point for a home's temperature and the mode of operations as well. Conversely, with a smart thermostat, the user may input the range of temperatures they would like a home to be at different times of day, to which the device creates an adjustment schedule that saves the user money from energy bills and without further input adjusts the temperature as instructed. The applications of IoT expand to encompass several industries generally integrated to display the status of "smart" (networked) devices via real-time feedback.

IIoT is much different. There is similarity in the effects of using IIoT devices: increased accessibility of data and increased data throughput. In short, the purpose of IIoT is meant to help other machines rather than another person, mainly focusing on machine-to-machine connectivity. One use case for this is if a company has related factories that produce a given part. If one factory has a shortage in its product, the other factory should also respond to this information and reduce the amount produced to prevent waste. With IIoT developments, this is now a possibility. Data is able to be aggregated from both facilities and transmitted to each other, and the IIoT devices can automatically adjust parameters according to the data.

1.2.2 How are IIoT Systems Deployed. According to [3], there is little standardization of how IIoT systems are setup, but most will recognize the separation of IIoT systems into the following four layers: perception, network, processing, and application. The first layer is primarily composed of sensors and actuators. In an IIoT system, the sensors are critical to providing information to any sort of automation solution as they provide the data that is needed to set controls through the actuators. The IIoT system also requires a method of communication, which is where the networking layer comes in. To handle the vast amount of data being aggregated, the processing layer helps out. IIoT devices handle the large sum of data through usage of cloud computing to handle heavier software such as various AI models or predictive modeling. Finally, the application layer is where all of the decisions made get relayed. In this layer certain triggers may exist based on the data processed, such as setting alarms, changes to an actuator, or other interactions with devices based on the data.

One example of this technology being applied is provided by MachineMetrics, a company who specializes in deploying IIoT technology. According to [4], MachineMetrics deployed IIoT monitoring technology for BC Machining. In order to decrease the downtime of their manufacturing machines and to aggregate data for adjusting the machine's operation. According to the case study, the deployment of predictive monitoring IIoT solutions greatly improved the capabilities for predicting tool breakages and problems before they occur using the data and predictive analysis. Many other IIoT companies produce solutions with similar purpose and goals in the manufacturing field. But as mentioned earlier, there is a great variation of needs from different companies and especially different sectors. Not only are the inherent goals from deploying IIoT devices different, but also the technology needing connection differs greatly.

1.3 Project

This project seeks to better protect the OT network environment by analyzing the learning capabilities of machine learning models, using data geared for both IT and OT networks to ensure generalization of data, and determine how many data points are required to let the model become consistent. Since many OT networks have different proprietary hardware, software, services, and protocols, many attacks may be specialized for these proprietary systems and therefore data may not be available. This would mean the only way for a model to learn about a new specialized attack would to be unfortunately affected by the attack first to get data on it. Since SCADA systems control many expensive processes, safety systems, and other critical infrastructure, it is extremely important that the amount of data required to learn a new attack is minimized, even at the cost of overall accuracy. Especially since many attacks which are benign to the IT infrastructure, such as port scanning, possess the ability to disrupt or delay communications within the OT infrastructure. This means that the IDS should be geared to protect against all levels of attacks, even those categorized as low threat in an IT infrastructure.

2 RELATED WORK

2.1 Machine Learning Intrustion Detection

One method that is frequently used to protect OT environments is by using rule based Intrusion Detection Systems (IDS), and more recently in research, machine learning based IDS systems specialized for anomaly detection of network traffic[5] [6] [7]. These devices sit on the OT network as its own device, and also have software installed on each compatible device to monitor the traffic across an OT environment[5]. The detection mechanisms vary on the specific device or software deployed. These rely on features such as signatures, patterns found by anomaly detection models, common attack behavior, similar traffic matching to a CVE or CWE, and other discerning features [5] [6] [7]. These are the main forms of IDS seen in real world deployment, either relying on classical algorithmic matching with scanning for signature patterns within communications similar to how antivirus software scans files, or by applying machine learning algorithms to identify anomalies within network behavior. Before the burst of machine learning oriented research, rule based and signature based IDS was the main avenue for real world deployment, but machine learning and deep learning have been expanding greatly as of recent. Although rule based IDS systems are not new in research, with Modbus TCP being a more recent standard, a more specific approach looks at Modbus TCP traffic with the goal of identifying misuse of certain Modbus commands using defined rule sets [8]. Modbus is one of the main communication protocols between SCADA devices, communicating raw input and output data to and from controllers and sensors. Modbus TCP was a recent development, bringing the raw signal protocol to ethernet. The goal with this work lies with the architect knowing the correct behaviors of the different devices on the network and setting up rules to ensure contrary commands are flagged with warnings or alarms. PCA based models have been applied to IDS and displayed promising results at gathering anomalous packets with high accuracy [9]. PCA models work by reducing dimensionality of the data and using learned weights to attempt reconstruction. Anomaly detection is performed this way by determining if the accuracy of a reconstruction significantly drops on a new piece of data, since in this case it would indicate an anomaly in the data causing the reconstruction to differ since the model is trained on. In this regard, by analyzing flow and packet behaviors on a network, [9] found success using this method to point out potential attacks by addressing anomalies within the data.

Others have seen success with employing long-short term memory nodes in deep networks [10], or employing ensemble deep learning models [11]. The perk of deep learning in general is it allows learning of less explainable patterns within the data, for networks in particular there may be certain patterns to regular communications that malicious ones do not have or vice-versa that would not be noticible to a human. LSTM models build off of this by also taking context into consideration, carrying over previous chronological data to make a determination about the current data point. Since attacks rarely only send one packet, this is an especially helpful technique to use with raw packet based data. The downside to both of these approaches is the overall cost it requires to run and train these models. Especially if the user wants their model to be very accurate, many layers of many nodes are required, which require high powered GPUs or TPUs to run at a reasonable speed. Many companies deploy models such as these on cloud servers where they can rent out exactly the amount of compute power they need. Whereas OT networks do not safely have such a luxury, since bidirectional communication is significantly discouraged between the OT network and outside sources, an optimal solution would require on-site technology. Most of these works target IT infrastructure and general deployment circumstances, likely due to the more niche nature of OT networks and the lack of real-world feasibility of deployment of large scale models within such an environment. With the unique requirements and behavior of OT networks, there is a gap towards creating specialized defenses for this type of network.

2.2 Datasets

There are many datasets that have been developed for general purpose anomaly detection, such as KDD Cup'99 [12] and its updated counterpart NSL-KDD][13]. Both of these datasets are based on communication flows rather than individual packets, allowing a clearer temporal sense of the packet data without requiring an LSTM or other temporal-based machine learning model. As previously discussed, not only is the OT environment different in composition, structure, and needs, but so too are the attacks different against the network. While being susceptible to more attacks than the IT network, attackers generally will focus on command and control of the network rather than individual systems holding critical data, internal denial of service to disrupt internal operations of the network, and transforming the many controller devices into a botnet, among other specialized forms of known attacks.

To ensure the model specifically works with OT networks, on top of using amplified datasets designed for general purpose anomaly detection, we also deploy a dataset geared for OT networks, ICS-Flow [14]. ICS-Flow is another flow-based dataset, which allows for direct performance comparison between the general purpose datasets and the OT specific dataset. Through the contrasts and potential harmonization of results between these datasets, greater insight can be had towards how a deployed model would generalize for deployment on an OT network.

It is therefore important that any IDS deployed on the network is specific to the environment, minimizing false positives to prevent loss of critical communications, and specialized to minimize the amount of data points required to learn a new attack. It is also therefore important to be able to generalize how quickly the model is expected to learn a new attack.

3 INTELLECTUAL MERIT

This project seeks to amplify the current landscape of machine learning based intrusion detection by not focusing on overall accuracy, but rather the learning rates of models. With new attacks becoming more and more frequent, it is important that any deployed model is able to stay accurate and adapt to new threats as they come, not just be accurate. Most of the research towards intrusion detection is focused on high accuracy and do not focus on learning new attacks fast. Especially in an OT network, where many services are proprietary, many operating systems are unique, and network communication can look vastly different between different organizations and their needs. Because of this, a generalized set of data may not be fully applicable to cover the attack landscape of organizations, thus it is especially important for an IDS deployed on an OT network should be able to learn rapidly to new data as it is likely that the attack may be specialized for their hardware and software vendors, of which there are many.

To accomplish this our project proposes an amplification to traditional, flow-based datasets, as well as tests with OT geared datasets to determine the capabilities for a model to learn new attacks for the OT network. In particular, we developed datapoints for the KDD'99 Cup[12] and NSL-KDD [13] datasets, introducing a cryptominer deployed on an embedded system of a simulated network. To ensure that this amplification was representative of the attack within the contexts of the datasets, it was analyzed against the other normal datapoints to ensure that the environment did not create any particular unique modifications to the flow compared to what the devices would look like in a real situation, and found that because the communications are tracked via flows of network traffic rather than individual packet headers, the data can be found to be representative because the flows anonymize the packets, creating representations of connections rather than individual clients and making the simulated environment irrelevant; only the actual communication patterns matter in this style of data.

4 METHODOLOGY

The selection of datasets spanned from inclusions of IT oriented datasets as well as OT oriented datasets. The inclusion of both KDD'99 Cup [12] and NSL-KDD [13] serves the purpose of covering common IT attacks, which an exposed OT network would also be susceptible to. Especially since both include various probing-based attacks, which are known to be especially disruptive to SCADA devices, sometimes even forcing a full disruption of activity. With the novel inclusion of the cryptomining attack, it helps make these datasets more representative of common attacks the OT networks face, as the problem of botnet based attacks increase within the OT network landscape.

As previously stated, we also include ICS-Flow [14] to ensure coverage of OT specific behaviors, both for normal operation and attacks. This dataset being another flow-based dataset allows for direct comparison between the other two dataset performances and this one. Insight can be gained towards if there is some significance in any performance gaps between the datasets for the same model, and if certain models are able to perform better with differing amounts of data points between the datasets. The potential for some models to perform better on the general datasets and another performing better on the SCADA dataset makes it important to analyze any differences when discussing deployment of an IDS on the OT network infrastructure.

For model selection we compare Random Forest, Decision Trees, and Naive Bayes models. These models were chosen for their relatively low hardware complexity requirements. Since devices in the OT environment are built to be robust instead of high-powered, the emphasis for devices is that they withstand time and new developments along with wear and tear, and thus are rarely replaced. Ideally, a solution geared for OT would have the same design principles in mind, such that the requirements to run the given IDS can be implemented on pre-existing servers or be able to run on lightweight embedded system servers on the network for years without needing to be replaced.

This testing consisted of an incremental addition of training data points per attack on each model, where each attack was treated as new. By isolating a single attack as new for each run of testing, it is possible to see differences in learning rates and understand how different attack approaches could be seen as more or less complex than one another. A random forest model was fitted to the entire data set absent from the data set A that classified as an attack a. Increasingly, the training data pool receives data points from A and fits a new model to the data. Each iteration of data point additions was tested on multiple seeds such that the metrics would not be significantly affected by the different splits chosen as the amount of data points used to train the model increased.

Data derived from this testing include the mean model accuracy, precision, and f-score for each amount of data used on the model. In addition, binary recall score is observed by creating a binary mask on testing data points that have classification a to observe the success in detecting a new attack as it is being learned. The evaluation data provides insight into the model's ability to generalize to novel attacks, while simultaneously ensuring high performance of the previously encountered attacks.

5 RESULTS

Random forest models demonstrated the fastest learning rate across all evaluated datasets as seen in Table 1 and as further shown in Figure 1, Figure 2, Figure 3 and Figure 4. The model's performance improved rapidly even with small amounts of training data. To reach 90% accuracy, the model required an average of 2 samples, with the slowest cases needing no more than 8 samples. This shows that the model can form a reliable decision boundary early, even when attack examples are limited.

Table 1: F1 Scores for all models used across increments of 20 data points.

Data Points	RandomForest	DecisionTree	Bayes
20	0.9698	0.9644	0.2616
40	0.9792	0.9735	0.2713
60	0.9859	0.9798	0.2624
80	0.9869	0.9807	0.2676
100	0.9860	0.9820	0.2368
120	0.9890	0.9846	0.2648
140	0.9895	0.9872	0.2343
160	0.9904	0.9886	0.2666
180	0.9899	0.9861	0.2577
200	0.9896	0.9883	0.2708

Performance at the 95% accuracy threshold showed similarly strong efficiency. The model needed an average of 6 samples to reach this level, with a maximum of 8 samples

Table 2: Average random forest model performance grouped in 2-point increments up to 20 data points.

Data Points	Recall	Accuracy	Precision	F1
2	0.2456	0.8812	0.9565	0.8821
4	0.4673	0.9044	0.9631	0.9114
6	0.6494	0.9314	0.9730	0.9409
8	0.7224	0.9451	0.9790	0.9534
10	0.7337	0.9454	0.9781	0.9531
12	0.7411	0.9462	0.9781	0.9534
14	0.7655	0.9493	0.9799	0.9566
16	0.7751	0.9510	0.9803	0.9581
18	0.7768	0.9513	0.9804	0.9582
20	0.7780	0.9514	0.9804	0.9583

Table 3: Average random forest model performance grouped in 20-point increments across all data points.

Data Points	Recall	Accuracy	Precision	F1
20	0.8456	0.9621	0.9859	0.9698
40	0.9057	0.9738	0.9893	0.9792
60	0.9324	0.9824	0.9912	0.9859
80	0.9353	0.9833	0.9920	0.9869
100	0.9434	0.9819	0.9926	0.9860
120	0.9503	0.9865	0.9933	0.9890
140	0.9585	0.9867	0.9940	0.9895
160	0.9606	0.9881	0.9941	0.9904
180	0.9618	0.9869	0.9941	0.9899
200	0.9631	0.9868	0.9939	0.9896

in the most difficult runs. The modest increase in required samples from the 90% to 95% threshold suggests that the model refines its classification boundaries quickly after initial convergence.

Reaching 99% accuracy required more data points across the board. On average, the random forest reached this threshold with 28 data points, and the most challenging attack type required 68 data points. This behavior is expected, as extremely high accuracy typically depends on higher data points to learn the more subtle features from a class. Even so, the model's ability to approach near-perfect accuracy with relatively few samples reinforces its suitability for deployment within an IDS on the OT network landscape where specialized attacks need to be contained fast.

When observing the ICS-Flow model in particular, there was a slight drop in initial accuracy across both random forest and decision tree models in comparison to the IT datasets. The OT specific attacks converged at the same rate as the IT models, needing around 8 data points to start achieving diminishing returns and plateaued around 120 data points.

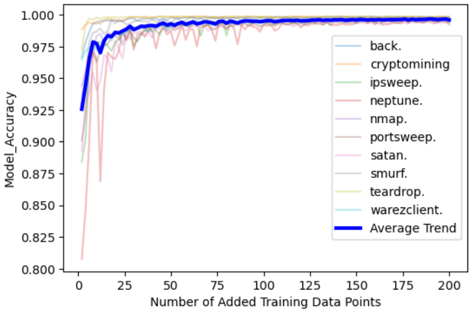

Figure 1: Accuracy of the Random Forest model as data points were added.

Using the full training sets, the random forest achieved 99.46% accuracy on the augmented KDD'99 dataset, 99.68% accuracy on the augmented NSL-KDD dataset, and on the ICS-Flow dataset a 98.12% accuracy, demonstrating consistent strength across the datasets. With the similar growth rate and the high accuracies, the random forest model performs significantly well and demonstrates the ability to learn new attacks within a fast pace and a minimal amount of attack occurances.

Looking at Table 2 and Table 3, we can see that most of the performance gain occurs within the first 8 data points, from there the returns diminish until the F1 score hits a plateau at around 120 data points. This occurance demonstrates the ability for the model to quickly learn the attacks with minimal data points, the additional data points past 8 serve to further increase its accuracy by small margins, but most of the learning is complete by around 8 data points. With the small margin of attacks required to mostly learn the attack, random forest is feasible for quick deployment within an OT network, requiring as little as one attack instance to generate the required data points. Since each data points is a single TCP connection flow, and the attacks covered span over more than one connection over bursts of time, a single attack can generate many data points, of which minimal are required to achieve above 95% F1 score. F1 score is used here to determine the feasibility due to the balanced nature of the statistic, ensuring that false positives are punished equally to false negatives. This is because if too many false positives occur, operators can ignore any triggered alarms, and false negatives mean that an attack goes undetected until damage occurs, at which point operators cannot prevent harm.

This would point towards random forest being the best contender for deploying a fast learning, fast acting model for an OT network. Another benefit of deploying the random forest algorithm as the chosen IDS machine learning algorithm would be that its overall training speed and prediction speed is fast and requires lower hardware requirements than

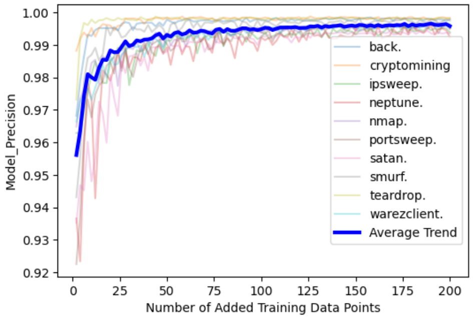

Figure 2: Precision of the Random Forest model as data points were added.

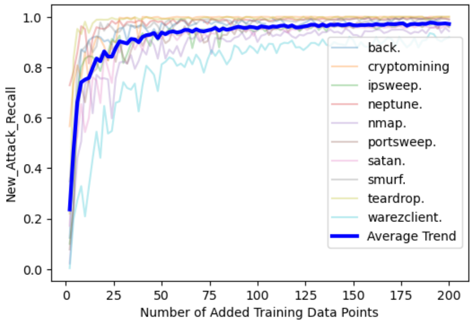

Figure 3: Recall of the Random Forest model as data points were added.

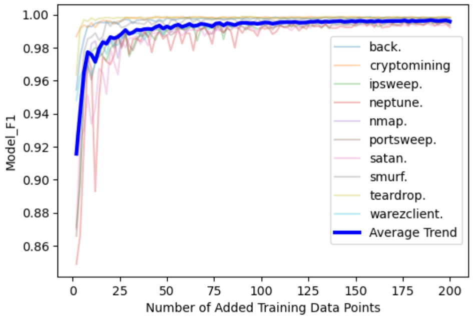

Figure 4: F1 score of the Random Forest model as data points were added.

other traditional machine learning models and deep learning models. Since SCADA infrastructure generally runs on older and robust hardware, these lower system requirements for deployment would make this form of IDS more accessible and deployable in a real world environment.

With the ability to reach above 95% accuracy with so little data points in both the average case and worst case, the random forest classifier was able to learn within 8 data points to that accuracy. For any extended attack, 8 data points, with each being a different connection, this would be an expected from a single attack. In other words, from a single attack, random forest is able to learn an attack with 95% accuracy regardless of the attack type.

In such an environment where this solution is deployed, the outputs of the IDS can be aligned with the alarm systems within the SCADA controller, flagging various levels of alarm depending on the determined threat level of each type of attack. Because the model setup we have, where attack types are separated and identified instead of a binary classification of attack present or no attack present, the manual classification of each attack type into various levels of alarm can be done. Furthermore, in a real world deployment of random forest as the chosen machine learning algorithm, the training process can be done on a separate, dedicated device by cybersecurity personnel, with the new, labeled data being loaded into the set of data used in the deployed model. The weights and splits can be saved and loaded onto the actual model to update the weights without disrupting the IDS scanning process or encumbering any devices on the OT network during retraining.

6 FUTURE WORK

With the growing hardware advancements and ability to process more and more complex models, future work can determine the learning abilities of deep learning models that require high computation power. This research relied on models with a relatively low hardware strain and requirement, but expanded breadth regarding the learning rates of other models would be beneficial. On top of this, recreation of other developed models with an emphasis on determining the learning rate and capabilities would provide further insight into this issue.

Since so many attacks are being developed for OT environments due to the vastly different communications needed to communicate across various brands of SCADA hardware, the multitude of proprietary protocols and services, and different vulnerabilities present determined by the model of hardware or software deployed, future work can analyze if there is an upper bound on the amount of attacks a given model can learn before accuracy and learning rates drop. This would provide unique insight towards how robust various models can be before they start breaking down.

The topic of model selection requires further consideration, as modern deep learning network packages are created with the ability for retraining in mind. Given that there is a steady in-stream of data (which is characteristic of OT networks), neural networks can change their objective between inference and continual learning. This is distinct from the shallow learning capabilities presented with random forest models, where a new training dataset would require refitting the model. This approach significantly reduces computational overhead, allowing for more agile model deployment into OT networks with minimal downtime.

7 CONCLUSION

This project sought to determine which model would be best fit for deployment on an operational technology network containing SCADA devices. Our chosen metric was the learning rate of attacks due to the frequency and potential need for attacks to be specialized for the OT environment and the chosen models and brands of devices used, and the lack of data present on those attacks due to their specialized nature. Because of the inherent real world danger present with each attack on OT devices, emphasis on learning rate over other accuracy and precision metrics that most research emphasizes, the learning rate helps identify and minimize the amount of attacks required to learn a new attack type introduced onto an environment.

When focusing solely on the learning rates, random forest showed clear improvement in learning newly introduced attacks. It reached 90% accuracy with an average of just 4 training samples, and even in the slowest cases required no more than 8 samples. The model continued to refine its decision boundaries quickly, achieving 95% accuracy with an average of 8 samples and a maximum of 10 samples. To reach 99% accuracy, the model required more data—averaging 28 samples and needing up to 68 samples for the most challenging attack type, but still converged far faster than the other models. Using the full training sets, random forest achieved 99.46% accuracy on the augmented KDD'99 dataset, 99.25% accuracy on the augmented NSL-KDD dataset, and 98.12% on ICS-Flow, demonstrating both rapid learning and strong overall performance across all evaluated datasets.

With the low hardware requirements of the random forest model for both training and active deployment, the random forest model is especially suited for the robust nature of OT network devices. Through the robust selection of datasets suited for general network environments and OT networks in particular, it can be shown that the learning rate results would be representative of what can be expected during a breach of the DMZ into the OT network, and the attacks experienced thereafter.

REFERENCES

[1] N. Jazdi. 2014. Cyber Physical Systems in the Context of Industry 4.0. In *2014 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*. Cluj-Napoca, Romania, 1–4. https://doi.org/10.1109/AQTR.2014.6857843

[2] M. Fagan, K. N. Megas, K. Scarfone, and M. Smith. 2020. *Foundational Cybersecurity Activities for IoT Device Manufacturers.* Technical Report NIST IR 8259. National Institute of Standards and Technology, Gaithersburg, MD. https://doi.org/10.6028/NIST.IR.8259 pp. iv–vi, 1-2, 17, 24.

[3] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund. 2018. Industrial Internet of Things: Challenges, Opportunities, and Directions. *IEEE Transactions on Industrial Informatics* 14, 11 (2018), 4724–4734. https://doi.org/10.1109/TII.2018.2852491

[4] G. Immerman. 2021. MachineMetrics Predictive: A Case Study Interview with BC Machining. (2021). https://hubs.ly/H0Fh4Jf0

[5] A. Borkar, A. Donode, and A. Kumari. 2017. A Survey on Intrusion Detection System (IDS) and Internal Intrusion Detection and Protection System (IIDPS). In *2017 International Conference on Inventive Computing and Informatics (ICICI)*. Coimbatore, India, 949–953. https://doi.org/10.1109/ICICI.2017.8365277

[6] A. Kiran, S. W. Prakash, B. A. Kumar, Likhitha, T. Sameeratmaja, and U. S. S. R. Charan. 2023. Intrusion Detection System Using Machine Learning. In *2023 International Conference on Computer Communication and Informatics (ICCCI)*. Coimbatore, India, 1–4. https://doi.org/10.1109/ICCCI56745.2023.10128363

[7] Z. S. Malek, B. Trivedi, and A. Shah. 2020. User Behavior Pattern - Signature Based Intrusion Detection. In *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*. London, UK, 549–552. https://doi.org/10.1109/WorldS450073.2020.9210368

[8] F. Katulić, D. Sumina, I. Erceg, and S. Groš. 2022. Enhancing Modbus/TCP-Based Industrial Automation and Control Systems Cybersecurity Using a Misuse-Based Intrusion Detection System. In *2022 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*. 964–969. https://doi.org/10.1109/SPEEDAM53979.2022.9842239

[9] Fernando Pérez-Bueno, Luz García, Gabriel Maciá-Fernández, and Rafael Molina. 2022. Leveraging a Probabilistic PCA Model to Understand the Multivariate Statistical Network Monitoring Framework for Network Security Anomaly Detection. *IEEE/ACM Transactions on Networking* 30, 3 (2022), 1217–1229. https://doi.org/10.1109/TNET.2021.3138536

[10] Arvin Hekmati, Jiahe Zhang, Tamoghna Sarkar, Nishant Jethwa, Eugenio Grippo, and Bhaskar Krishnamachari. 2024. Correlation-Aware Neural Networks for DDoS Attack Detection in IoT Systems. *IEEE/ACM Transactions on Networking* 32, 5 (2024), 3929–3944. https://doi.org/10.1109/TNET.2024.3408675

[11] Chaoyun Zhang, Xavier Costa-Pérez, and Paul Patras. 2022. Adversarial Attacks Against Deep Learning-Based Network Intrusion Detection Systems and Defense Mechanisms. *IEEE/ACM Transactions on Networking* 30, 3 (2022), 1294–1311. https://doi.org/10.1109/TNET.2021.3137084

[12] Santhosh Kumar B.J. 2025. KDDCup99. (2025). https://doi.org/10.21227/v5ch-eh64

[13] RUIZHE ZHAO. 2022. NSL-KDD. (2022). https://doi.org/10.21227/8rpg-qt98

[14] Alireza Dehlaghi-Ghadim, Mahshid Helali Moghadam, Ali Balador, and Hans Hansson. 2023. Anomaly Detection Dataset for Industrial Control Systems. *IEEE Access* 11 (2023), 107982–107996. https://doi.org/10.1109/ACCESS.2023.3320928