Hybrid Cyber-Physical Intrusion Detection System for Smart Manufacturing

Jeremy Potts and Muhammad Ismail
Department of Computer Science, College of Engineering, Tennessee Tech University
Cookeville, TN, USA
Emails: {jpotts, mismail}@tntech.edu

Abstract

Smart manufacturing is an important part of our critical infrastructure and is the current age of industry where physical components such as robotic arms, 3-D Printers, CNC machine, etc. are all interconnected and remotely controlled or automated which provides a major boost in efficiency.[9] While these systems are more convenient, the cyber-physical integration expands the attack surface of these systems for any potential threats to act on and exploit. This integration also creates gaps in the current intrusion detection systems (IDS) and the research of such systems as they focus on either the cyber or physical components of these system which leaves blind spots when an attack can only be detected by using either cyber or physical data. This paper presents an experiment conducted to fill that gap by creating a cyber-physical testbed, launching Denial of Service and Physical Hijacking attacks, collecting benign and malicious data, and creating a hybrid IDS utilizing K-Nearest Neighbors and Decision Tree models that consider both cyber and physical data. This hybrid IDS produced good results, achieving an accuracy of 97.2% which is roughly the same as separate cyber and physical IDSs, but there is a significant boost in precision (98.4%), recall (94.2%), and F1 score (96.1%) when using the hybrid IDS compared to the separate IDSs.

Introduction

Smart manufacturing is the latest step in advancing the manufacturing industry, where most components that are manually operated can now be done autonomously and controlled remotely. Common smart manufacturing systems would include machines like robotic arms, precision cutters, conveyor belts, 3-D printers, etc. These systems use Programmable Logic Controllers (PLCs), Human Machine Interfaces (HMIs), and networking to create a cooperative environment where machines of similar and different types work together with human interaction to create a complete production system.

Smart manufacturing systems are considered to be a part of our critical infrastructure meaning that they are crucial to our day-to-day operations. As part of our critical infrastructure, if something where to go wrong within these systems, it could cause a great amount of damage to both physical infrastructure and human life. In the past few years, we have witnessed a rise in attacks on these critical infrastructures that led to devastating effects. In 2021, there was a ransomware attack on Colonial Pipeline that led to the shutdown of a main gas supply line that ran through and supplied gas to the entire east coast of the United States.[10] The shutdown caused major disruptions and panic of citizens as well as major safety concerns as many citizens, in panic, began to mass purchase gas and store it in unsafe containers. In late 2023, there was an attack on the Pennsylvania water authority that led to a shutdown and eventual manual operation of a remote pumping station that regulates pressure to local towns.[11] This attack targeted specific PLCs with design flaws that led to the breach and affected the water flow to the local towns. This same attack was also executed across several other industries that utilized the same type of PLC. These recent attacks show that our cyber-physical critical infrastructure is vulnerable and currently available to exploit. As a result, there is an urgent need to develop effective intrusion detection systems (IDS) that can detect and mitigate these serious attacks.

Currently, the research in this field [1]-[8] is limited in its execution, meaning that it uses either cyber data or physical data when developing their IDS. These infrastructures are cyber-physical systems, which means that using only cyber or physical data leaves the IDS blind to half of the picture. This can lead to undetected attacks because they only manifest in either the cyber or physical space and are virtually invisible in the other space. To address this limitation, we propose in this paper a hybrid IDS that fuses cyber and physical models to effectively detect attacks on cyber-physical systems. Our contributions in this paper can be summarized as follows:

- We develop a cyber-physical testbed that mimics a smart manufacturing facility with robotic arms. Then, we launch two attacks that manifest in the cyber domain or in the physical domain, and collect benign and malicious data needed to develop an IDS.
- We develop an ensemble machine learning model for the IDS based on cyber-physical fusion utilizing K-Nearest Neighbors and Decision Tree models.
- We conduct comprehensive evaluations of our developed IDS, which show that our model achieved the same ac-

curacy as classical models ($97\% - 98\%$) and improved precision (98%), recall (94%), and F1 score (96%).

The remainder of this paper is organized as follows. Section II reviews the related works and highlights their limitations. Section III details the components of the developed testbed and outlines the workflow of the system. Section IV details the type of datasets collected, the tools that are used to collect the data, the attacks that are executed, and the steps to pre-process the data for training. Section V details what machine learning models are used and how cyber-physical data fusion is achieved. Section VI reviews the results and discusses the findings. Finally, conclusions are given in Section VII.

Related Work

This section reviews the related works and highlights their limitation to motivate our work.

Cyber-Only IDS Research Cyber-only IDSs are trained solely on cyber data features. This type of data is collected from the various interactions/communications between machines. IP addresses, port numbers, sizes, payloads, etc. are typical features you would find in a cyber dataset.

In [6], a tool called C3PO is developed for security analysis to identify security threats for networked 3D printer. This tool is used to analyze 13 networked printers and five real-world network deployments. The work in [1] discusses the use of a state-based IDS for robotic arms that uses a hierarchical support vector machines based on the particle swarm optimization algorithm. The IDS uses real-time joint data to determine the state the robotic arm is in and if there is any anomalies. This paper also discussed an automated response to a detected incident. The work in [2] discusses the use of a Snort-BASE IDS for detecting DoS attacks on robotic arms. The described system uses Snort to record the network traffic between the arm and its controller and uses BASE to perform analysis on the recorded network traffic. This work also describes the tests performed on a robotic arm with six degrees of freedom in both a completely simulated environment and physical implementation.

Physical-Only IDS Research Physical-only IDSs are trained solely on physical data features. This type of data is collected from physical instruments or sensors that are placed in the physical world. Temperature, speed, location, pressure, etc. could be features you would find in a physical dataset.

The work in [4] proposes a new framework that incorporates Blockchain to secure a quality control application of the entire supply chain(supplier, manufacturer, and retailer). It also uses machine learning techniques and supply chain data to predict the quality of the product being produced. The work in [5] explores security of smart manufacturing systems and answers four questions: Under which conditions are certain attacks possible, what is their impact, are there any overlooked attack vectors, and what is the security impact of the current software-development practices. To answer these questions they set up an environment and ran four attack scenarios: HMI exploitation, compromise via 3rd-party firmware libraries, persistence and production alteration, and add-in or digital twin compromise. The work in [3] discusses the use of cameras in the working space of a robotic arm to detect its proximity to a human. It uses a combination of color and 3-D Time of Flight information to detect where the arm and human are and calculate the distance between them. The goal of this system is to prevent any collision between human and robotic arm to prevent injuries. The work in [7] proposes a scheme for intrusion detection for 3D printers through acoustic signature, real-time tracking of machine components, and post production materials analysis. They made a connection between the instruction given and the noise the resulting action made while printing. They use this connection as well as tracking where the machine components are supposed to be and what they are supposed to be doing to build an intrusion detection system. The work in [8] proposes a framework for intrusion detection that compares real-time side-channel signals. This framework also proposes a new tool that fixes time differences between measurements on different runs on the same system.

Limitations of Literature

There are two main limitations in the literature. The first is that there are several works that limit their testing to only a single type of attack. This limitation can lead to many blind spots in IDSs because they are used for all-round attack detection and if they cannot detect multiple types of attacks with high accuracy then they become obsolete. The first limitation leads to the second, which is that no research develops an IDS that utilizes cyber-physical data fusion. Cyber-physical data fusion is the creation of a new dataset that merges the cyber data and physical data, so a single instance will not only have cyber features like IPs, port numbers, protocols, etc. but also physical features like temperature, speed, position, etc. Cyber-physical data fusion provides a more holistic dataset that when used for training has the potential to build a more robust IDS that could detect multiple types of attacks that other detection systems would be blind to.

Testbed Details

To develop a hybrid cyber-physical IDS and test our hypothesis that cyber-physical fusion leads to better IDS, we developed a testbed that mimics a smart manufacturing environment. Our testbed has two main layers, the physical layer and the cyber layer. The physical layer consists of four components: one Trossen Robotics PX100 robotic arm[12], one Raspberry Pi 4, one Netgear GS308[13] networking switch, and a laptop. The PX100 arm is a robotic arm that has four degrees of freedom operated by five DYNAMIXEL XL430-W250 Servos[14] and controlled by a DYNAMIXEL U2D2 board[15]. The U2D2 controller board is connected to the Raspberry Pi 4 through a USB to Micro-USB connector. This Raspberry Pi 4 acts as a PLC and is running a lightweight operating system installed with ROS Noetic[16], Python, and custom Python wrappers provided by Trossen Robotics[17]. Each of these three softwares plays a crucial role in the operation of the arm. ROS Noetic facilitates

the connection and control signals to make the arm move. Python is used the create programs that call on ROS Noetic to change control signals. The custom Python wrappers provide easy-to-use API functions that make calling on ROS simpler. The Raspberry Pi has Wi-Fi disabled and is also connected to a laptop through an Ethernet connection via the unmanaged Netgear GS308 switch.

The second layer, the cyber layer, of this testbed consists of four components: a controller, an HMI, a logger, and an attacker. All four of these components are housed in virtual machines (VM) installed on the laptop. The controller, HMI, and logging VMs are all running Ubuntu 16.04 and ROS Melodic[16]. These three machines are connected to each other and the physical system through a NAT network and each machine also serves a specific purpose in the testbed. The controller VM is used to remotely communicate with the Raspberry Pi and control the arm. It does this using custom python scripts that load programs onto the Pi and remotely execute them. The HMI VM is used to remotely monitor the movements of the robotic arm in real-time by utilizing ROS communications and Rviz to provide a graphical view of the real arm. The logging VM is used to collect the data from both the physical arm and network/cyber data from the entire testbed. For physical data collection, the machine utilized a ROS communication package to pull joint state data from the arm. For networking/cyber data collection, the machine uses Wireshark to capture all network packets in the testbed. The final VM in the cyber layer is the attacker box. This VM is running Kali Linux and is used to simulate a malicious machine, compromised or otherwise, on the system to disrupt operations.

The overall workflow of the testbed is as follows. First, the Raspberry Pi and robotic arm are powered, the Pi startup script is run to connect to the arm and it waits, listening for commands to run. Second, the monitoring VM is connected to the Pi to begin monitoring the arm in real time. Third, the logging machine begins logging both physical joint states and network data. Finally, the control machine connects to the Pi and sends a program and executes it making the arm perform desired functions. The final step can be repeated as many times as necessary.

Data Collection

As discussed in the previous sections, there are two layers of the testbed and we want to collect data from both layers to study advantages and disadvantages of using cyber-physical data fusion when detecting various types of attacks. To accomplish this goal, two different tools are used. The first tool used is Wireshark to collect all network traffic packets traveling across the network of the testbed. Hence, the cyber data collected from the network are Packet No., Time, Source IP, Destination IP, Protocol used, Length of packet, and Payload of the packet. The second tool used is RosTopic, which is a data analytic software package from ROS that allows for the collection of joint states of the robotic arm. Each instance recorded contains three sets of seven measurements taken from the arm. There is one set of position measurements where the current orientation of the five servos and two fingers of the arm are recorded. There is one set of effort measurements where the current amount of energy used to hold or move the five servos and two fingers of the arm are recorded. Finally there is one set of velocity measurements where the current speed of the five servos and two fingers are recorded. In total, each instance will have 21 data features that are recorded as floating point values.

Benign Data

When trying to analyze data to detect attacks, there are two types of data that need to be collected from the system. The first type of data needed is benign data or data collected under normal operation of the system. To simulate normal operation we programmed the robotic arm to repeatedly perform a simple action of picking an object up, moving it, placing it down, and moving back to a neutral resting position. In the benign simulation, we made sure to follow the workflow described in the previous section in order to collect a good sample of usable data. In total the number of packets captured is $205,043$ and the total number of physical instances from the arm is $194,738$.

Malicious Data

The second type of data needed when detecting anomalous behaviour is the malicious data. For this experiment, the malicious data is generated as a result from performing various types of attacks on the system. This experiment only featured two types of attacks specifically chosen to test the different capabilities of a cyber-only trained detection system, a physical-only trained detection system, and a hybrid cyber-physical trained detection system. The first type of attack featured in this experiment is a physical attack simulated as a local hijacking attack. This attack was executed by locally loading and executing a malicious job on the PLC that would move the arm in unexpected patterns. The attack bypasses the normal workflow and interrupts the currently running benign job to run the malicious job. This type of attack is chosen because it would have the best chance of going undetected by a cyber-only trained detection system. The second type of attack featured in this experiment is a cyber Denial-of-Service (DoS) attack against the monitoring VM. This DoS attack is a SYN-Flood attack where the HMI VM is sent a very large number of new connection requests over the same port that the Rviz HMI is communicating with the PLC. The flood of new connections slows down or completely blocks the communication to the PLC, which makes the Rviz HMI slow and inaccurate. This attack is chosen because it would be very obvious for any cyber-only trained detection system but would be virtually impossible to detect using a physical-only trained detection system. The way that data is collected for both of these attacks is the same. The system is booted as per the workflow described. Once the system is running a benign job, then either of the attacks are executed, data is collected, and the system is shutdown. In total there are $40,205$ malicious packets captured ($20,205$ from the hijack attack and $20,000$ from the DoS attack) and $22,098$ malicious instances from the arm ($3,228$ from the DOS attack and $18,870$ from the hijack attack). There are

no measures taken in this paper to compensate for the data imbalance between the benign and malicious datasets.

Data Pre-processing

Both benign and malicious datasets require a substantial amount of data pre-processing. There are two different sets of steps taken when dealing with either cyber data or physical data no matter if the set is malicious or benign. The cyber datasets are originally stored as PCAP files and once converted to a CSV by Wireshark, each string and non-standard numerical value needs to be encoded and transformed into a standard numerical value. The data fields that are encoded are Source IP, Destination, IP, and the Packet Information fields. To eliminate any possible unintended and unrelated connections, the Packet Number and Time data fields are dropped from the dataset. The physical datasets are originally stored as normal text files containing each instance in the form of a packet with header information and the data itself. In order to convert them into CSV files, the header information for each instance is removed as well as unneeded text titles. Once both malicious and benign datasets for both cyber and physical data are converted into CSV files, they are all merged into one dataset for their respective source (cyber or physical) and given a label, zero for benign instances and one for the Dos attack and two for the hijack attack.

IDS Training

Due to the way the data is collected, there is no logical connection between the physical dataset and the cyber dataset. This means that classical cyber-physical data fusion could not be achieved. To still achieve the effect of cyber-physical data fusion, we use an ensemble learner of learners that are trained on the cyber and physical data separately. The ensemble learner consists of the K-Nearest Neighbors (KNN) and Decision Tree (DT) models. K-Nearest Neighbors is the first machine learning model chosen because of its natural logical connection to position data of the robotic arm. Decision Tree is the second machine learning model chosen because it is able to make better connections with the diverse feature space of the cyber data.

There are three stages of training for this experiment. First, both of the machine learning models are trained using the physical data and cyber data separately using an $80/20$ split for training and testing. Then, the models are tested using their testing sets and the predicted values as well as the actual values for each instance are saved. Finally, an ensemble learner using KNN and DT models is trained and tested on a combined dataset of the predicted and actual values saved from both cyber and physical model testing. Because there is an imbalance between the size of the datasets, each instance of predicted and actual values in the physical dataset is used and it is combined with a randomly selected instance in the cyber dataset that shares the same actual value to form the cyber-physical dataset. Each instance of the cyber-physical dataset will have five features: the predicted values from the KNN and DT models trained on both cyber data and physical data, and the actual value.

Results and Discussion

The results of this experiment can be broken down into three parts, the performance of the cyber only IDS, the performance of the physical only IDS, and the performance of the cyber-physical IDS. The average accuracy, precision, recall, and F1 score for all three IDSs can be seen in Table I.

Table 1: Summary of average detection results

	Accuracy	Precision	Recall	F1 Score
Cyber-Physical	97.2%	98.4%	94.2%	96.1%
Cyber Only	97.5%	98.5%	90.1%	94.6%
Physical Only	97.9%	78.3%	69.95%	71.1%

As shown in Table I, the accuracy is fairly close between the three IDSs, and the main difference when comparing the three IDSs is in the precision, recall, and F1 score. When comparing the cyber-physical IDS to the cyber only IDS, we can see that they perform the roughly the same or the cyber-physical IDS performs a couple percentages better than the cyber only IDS. However when comparing the cyber-physical IDS to the physical only IDS we can see that there is a significant improvement in all three stats when using the cyber-physical IDS rather than the physical only IDS. When we compare the cyber only IDS to the physical only IDS we can see the the cyber only IDS out performs the physical only IDS in precision, recall, and F1 score.

There are two reasons for this significant performance difference. The first is the amount of data in the respective datasets. The cyber dataset had significantly more instances than the physical dataset because the rate of data collection for the physical data is slower than the amount of traffic that traveled across the network. The second reason for the performance difference is the contamination of cyber data with physical data. This contamination occurs because all the physical data is remotely collected, so this data is contained in network packets that Wireshark collected and the cyber-only IDS trained on. Because the IDS was able to see the physical data it is then able to able to detect attacks which otherwise would have been invisible.

Conclusion

The introduction of cyber-physical systems in our manufacturing process has provided a boost in both the efficiency and safety of the industry. The major drawback of this advancement is that it increases the attack surface for the system opening it up to more possible attacks that may cause harm to both infrastructure and people. The research of this problem has only been considering either cyber or physical data when training their intrusion detection systems, which leaves them blind to possible attacks in the other part of the system. This paper developed a cyber-physical testbed, attacked it using attacks that would manifest in either the cyber or physical space but not both, and developed a hybrid cyber-physical intrusion detection system that maintained accuracy and improved precision, recall, and F1 score compared to the classical intrusion detection systems.

References

Y. Zhou, L. Xie, and H. Pan, “Research on a PSO-H-SVM-Based Intrusion Detection Method for Industrial Robotic Arms,” Applied Sciences, vol. 12, no. 6, p. 2765, Mar. 2022, doi: 10.3390/app12062765.

Li, N. , Wang, Y. , Shen, P. , Li, S. and Zhou, L. (2022) A DoS Attacks Detection Aglorithm Based on Snort-BASE for Robotic Arm Control Systems. Journal of Computer and Communications, 10, 1-13. doi: 10.4236/jcc.2022.104001.

Robla, S., Llata, J.R., Torre-Ferrero, C. et al. Visual sensor fusion for active security in robotic industrial environments. EURASIP J. Adv. Signal Process. 2014, 88 (2014). https://doi.org/10.1186/1687-6180-2014-88

Z. Shahbazi and Y.-C. Byun, “Integration of Blockchain, IoT and Machine Learning for Multistage Quality Control and Enhancing Security in Smart Manufacturing,” Sensors, vol. 21, no. 4, p. 1467, Feb. 2021, doi: 10.3390/s21041467.

F. Maggi et al., ‘Smart Factory Security: A Case Study on a Modular Smart Manufacturing System’, Procedia Computer Science, vol. 180, pp. 666–675, 2021.

M. McCormack, S. Chandrasekaran, G. Liu, T. Yu, S. DeVincent Wolf and V. Sekar, "Security Analysis of Networked 3D Printers," 2020 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 2020, pp. 118-125, doi: 10.1109/SPW50608.2020.00035.

C. Bayens, T. Le, L. Garcia, R. Beyah, M. Javanmard, and S. Zonouz, ‘See No Evil, Hear No Evil, Feel No Evil, Print No Evil? Malicious Fill Patterns Detection in Additive Manufacturing’, in 26th USENIX Security Symposium (USENIX Security 17), 2017, pp. 1181–1198.

S. Liang, X. Peng, H. J. Qi, S. Zonouz and R. Beyah, "A Practical Side-Channel Based Intrusion Detection System for Additive Manufacturing Systems," 2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS), DC, USA, 2021, pp. 1075-1087, doi: 10.1109/ICDCS51616.2021.00106.

Thompson, Lauren. “What Is Smart Manufacturing, and How Is It Changing the Industry?” Engineering.tamu.edu, 14 Mar. 2022, engineering.tamu.edu/news/2022/03/what-is-smart-manufacturing-and-how-is-it-changing-the-industry.html.

Easterly, Jen, and Tom Fanning. “The Attack on Colonial Pipeline: What We’ve Learned & What We’ve Done over the Past Two Years | CISA.” Www.cisa.gov, 7 May 2023, www.cisa.gov/news-events/news/attack-colonial-pipeline-what-weve-learned-what-weve-done-over-past-two-years.

CISA. “IRGC-Affiliated Cyber Actors Exploit PLCs in Multiple Sectors, Including U.S. Water and Wastewater Systems Facilities | CISA.” Www.cisa.gov, 1 Dec. 2023, www.cisa.gov/news-events/cybersecurity-advisories/aa23-335a.

Trossen Robotics. “PincherX 100 Robot Arm - X-Series Robotic Arm.” Trossenrobotics.com, 2021, www.trossenrobotics.com/pincherx-100-robot-arm.aspx.

NETGEAR. “300 Series SOHO Unmanaged Switch - GS308.” NETGEAR, www.netgear.com/business/wired/switches/unmanaged/gs308/.

“DYNAMIXEL XL430-W250-T.” Www.trossenrobotics.com, www.trossenrobotics.com/dynamixel-xl430-w250-t.aspx. Accessed 29 Jan. 2024.

“DYNAMIXEL U2D2.” Www.trossenrobotics.com, www.trossenrobotics.com/dynamixel-u2d2.aspx. Accessed 29 Jan. 2024.

ROS. “ROS.org | Powering the World’s Robots.” Ros.org, 2020, www.ros.org/.

“Interbotix_ros_manipulators/Interbotix_ros_xsarms at Main · Interbotix/Interbotix_ros_manipulators.” GitHub, github.com/Interbotix/interbotix_ros_manipulators/tree/main/interbotix_ros_xsarms. Accessed 29 Jan. 2024.