Enhanced Intrusion Detection in Cyber Physical Systems using Graph Neural Networks

M. Maimoon Shirin[1], J. Priskilla Angel Rani[2]
[1]*Student, Francis Xavier Engineering College*
[2]*Asistant Professor, Francis Xavier Engineering College*

***Abstract*—A computer network may be impacted by hostile attacks, malicious software, and computer viruses. As a part of a defense technology, intrusion detection plays a major role in protecting the networks and its data. Poor accuracy, inadequate detection, a high false-positive rate, and an inability to handle novel incursion types are some of the problems that plague traditional intrusion detection systems. We suggest a unique deep learning-based technique to identify cybersecurity flaws and breaches in cyber-physical systems in order to address these problems. The suggested framework compares deep learning-based discriminative methods with unsupervised methods. This paper presents a promising solution such as Graph Neural Network (GNN) to detect network attacks in IoT-driven IICs networks. The results shows an 95% in terms of accuracy, reliability, recall, and efficiency in identifying intrusion attacks. For this paper, the output achieves the maximum accuracy rate in KDCupp99 datasets.**

***Index Terms*—Intrusion Detection, Graph Neural Network, Cyber Physical System, Cyberattacks, Malicious, Normal attack.**

I. INTRODUCTION

For CPS to identify and mitigate threats, intrusion detection systems, or IDS, are essential. Intrusion detection systems (IDS) are part of a system's subsequent protection line[1]. IDS is an observing system that detects suspicious activities and produces alerts when they are detected and implemented in conjunction with security concerns and procedures such as authentication, security system and encryption approaches to strengthen security against cyber-attacks. Employing a variety of benign traffic/ normal flow patterns and precise attack-specific rules, IDS can distinguish between harmful and non-malicious activity [5]. The recent technologies such as home automation, smart grids, automated vehicles, automated healthcare systems and industrial machines are made up of CPS [2] (Cyber-Physical Systems). These system are simply denoted as the combination of the cyber world with the physical world. Generally [3] the attacks are the common possible step in the CPS as it can cause damage to the systems, networks or securities to organization that adapts the CPS and majorly this leads to the financial destruction. However[1], the traditional intrusion detection systems are not capable to safeguard the networks from the cyberattack. The main complex of IDS is because of its predefined attack or basic anomaly detection methods. To manage all this Graph Neural Networks (GNN) [6], provides a best strategy. This GNN mainly understands the abstract and the underlying structure of the system to produce the best results. The main aim of the paper is to increase the protection of Cyber Physical System in modern era.

II. LITERATURE SURVEY

Some the of literature work for the enhanced intrusion detection system using GNN is,

Kandhro, Alanazi, Kanwal Fatima, and Mueen Uddin (2023), introduced an innovative deep learning method for detecting vulnerabilities and breaches in cyber-physical systems related to cybersecurity. Their framework contrasts deep learning-based discriminative techniques with unsupervised methods. To find cyberthreats in IoT-driven industrial control systems (IICS) networks, the study uses a generative adversarial network. This paper produces a results, that demonstrate an improvement in functionality with an accuracy, reliability and efficiency.

M. A. Umer, K. N. Junejo, M. T. Jilani, and A. P. Mathur (2022), introduced ML techniques for enhancing Industrial Control Systems (ICS) ability to protect the systems from cyberattacks. The method they introduced majorly focused on two areas. They are usage of network packets to identify

the unauthorized attacks at the network level, and finding out the difference in the physical system using the data that observed by the systems physical behavior. This paperwork highlights challenges in the application of these methods and offers recommendations to address them.

V. Ponnusamy, M. Humayun, N. Z. Jhanjhi, A. Yichiet, and M. F. Almufareh (2022), begin by reviewing various techniques for intrusion detection, methods of data gathering and placement tactics. The primary finding of this paper emphasizes how few network traces are available for training contemporary machine-learning models against intrusions unique to the Internet of Things. The KDDCup datasets is specifically examined to illustrate the design difficulties of wireless intrusion detection based on existing data properties. A number of recommendations are made to future-proof traffic capture techniques in the wireless network (WN). A overview of several intrusion detection, data collecting, and placement approaches opens the paper. This paper's primary objective is to examine the design difficulties associated with implementing an intrusion detection system in a wireless setting. Because of the architectural complexity, deploying an intrusion detection system in a wireless environment is more difficult than in a wired network context. In this study, the conventional wired intrusion detection deployment methods are reviewed, their potential adoption in the wireless environment is discussed, and the design constraints in the wireless environment are emphasized.

S. Layeghy, and M. Portmann (2021), presented four benchmark datasets and a NetFlow-based NIDS feature set, which they have made available to the research community. Network intrusion detection systems, often known as NIDSs, are crucial instruments for defending computer networks from more complex and frequent cyberattacks. Like any other ML-based application, ML-based NIDS requires high-quality datasets for training and evaluation. Because each publicly available datasets uses a unique and private set of attributes, it is practically impossible to compare the performance of ML-based traffic classifiers on multiple datasets. To solve that limitation, this paper proposes and evaluates common NIDS feature sets based on the NetFlow network meta-data collection system and protocol.

S. Sarhan, W.W. Layeghy, and Gallagher (2022), this paper is based upon Network Intrusion Detection System based on GNN. The datasets and the system will be trained and evaluated using GNN and represented in the form of Graph. The authors proposed the framework as E-GraphSAGE, which allows to capture both the edge features of the graph as well as topological information for the NIDS. This is the first model that has been created, implemented, and executed practically using GNN, but this also faces the challenges in NIDS because of the flow-based data.

M. Zong, M. Lin, M. Zang (2024), this paper proposes about the methodologies, trends and challenges related to Intrusion Detection System in GNN. The research mainly covers Graph Construction, Network Design and GNN models deployment. The generalization technique in the GNN based intrusion model addresses the challenges in the each stage. By adopting a problem-oriented taxonomy and conducting a targeted survey, this review aims to provide scholars with a clear, systematic framework for deepening their understanding and further exploration of the field.

These survey papers collectively provide a strong foundation for understanding the role of GNNs in threat detections within Cyber-Physical Systems, discussing both their potential and the challenges that remain.

III. PROPOSED METHODOLOGY

If The proposed framework contains a novel graph neural network (GNN)-based method for detecting cybersecurity vulnerabilities and breaches in cyber-physical systems. The proposed framework contrasts unsupervised learning methods with discriminative strategies based on GNNs. This study introduces a GNN-based model to detect cyberthreats in IoT-driven Industrial Internet of Things (IIoT) networks. The methodology used in Graph Neural Networks has the capability to find out the critical relationships and connections within the network data in order to capture the attacks or threats. During the training and the testing phase, this proposed approach guarantees the confidentiality and the integrity of users data and systems data.

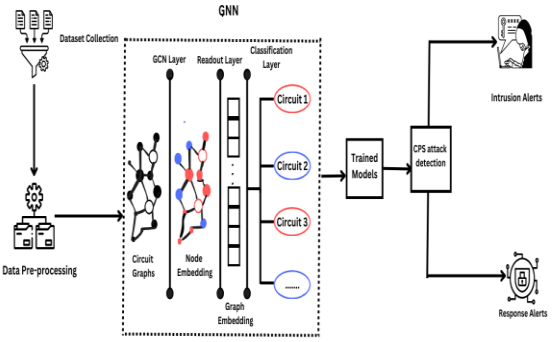

Figure 1: Architecture of the proposed system

Proposed GNN-based Detection Approach: A set of designed structure is adopted in the architecture for the Graph Neural Networks for Threat Detection in Cyber-Physiacal Systems provides an structured pattern, provides guarantee for the efficient attack detection in Cyber Physical environments. These proposed architecture contains some complex stages such as data collection, preprocessing, response mechanisms and training phases and so on. These important phases that make up the architecture.

Dataset Used: The first step in the design is gathering data to train the model. Here KDDCup99, NSL-KDD and UNSW-NB15 are the most popular and widely used datasets in terms of network attacks. As an extension of KDD99, the NSL-KDD datasets reduces the drawbacks of the previous version. This datasets totally contains 42 features and it can be classified into 3 sets such as traffic features, content features and content features. One of the most well-known IoT cybersecurity datasets is KDDCup 99. Together, PerfectStorm (IXIA) and the UNSW Cyber Range Lab created the UNSW-NB15 datasets, which generates assaults and acts that are fairly aggressive. The dataset contains 47 features per record, which are divided into 6 categories. Some of them are normal, neptune, smurf, portsweep, satan and teardrop.

src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	num_failed_logins	logged_in	num_compromised	...
2.519200e+04	2.519200e+04	25192.000000	25192.000000	25192.00000	25192.000000	25192.000000	25192.000000	25192.000000	...
2.433063e+04	3.491847e+03	0.000079	0.023738	0.00004	0.198039	0.001191	0.394768	0.227850	...
2.410805e+06	8.883072e+04	0.008910	0.260221	0.00630	2.154202	0.045418	0.488811	10.417352	...
0.000000e+00	0.000000e+00	0.000000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	...
0.000000e+00	0.000000e+00	0.000000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	...
4.400000e+01	0.000000e+00	0.000000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	...
2.790000e+02	5.302500e+02	0.000000	0.000000	0.00000	0.000000	0.000000	1.000000	0.000000	...
3.817091e+08	5.151385e+06	1.000000	3.000000	1.00000	77.000000	4.000000	1.000000	884.000000	...

Figure 2: Overview of collected dataset

Data Preprocessing: After the completion of collecting the raw data process, the system will undergoes the pre-processing phase to arrange the collected raw data into a suitable data format . So that the data can be used for modelling the GNN algorithm which in-turn produces the trained model. Some of the tasks that takes place in this step are:

Graph Neural Network (GNN): After preprocessing, the Graph Neural Network (GNN) will be applied. The basic motive of GNN is to manage and handle the graphical data because they are capable for capturing the intricate relationships. At this point the GNN framework contains many stages. The collected data will be modeled as a graph is the first step. This graph is called a circuit graph, where each node represents a physical or computational component of the CPS and edges describe the connections or interactions between these components. The graph structure helps capture the topology and communication patterns within the system. The second step is the Graph Convolutional Network (GCN) layer in which the data will be represented as graph and the main purpose is to gather the useful information from the circuit graph and this layer will perform the convolution operation by combining the data from the neighbor nodes. The third step is the Node Embedding and this mainly contains high-dimensional vector that contains the behavior of the node and its connection with other nodes. This embedding process is very much useful for intricate the network patterns. The fourth step is the Readout layer in which all the data in the graph will convert into a single vector and this process produces the global representation of the CPS network which is known as graph embedding. The fifth step is the graph embedding that holds the single vector that contains the information about the entire CPS network and at this stage the classification process will take place. The classification step is the final stage which is mainly used to predict whether the system is functioning normally or not.

Trained Models: Once the GNN model is trained using datasets, it can generalize into new sets of data. The trained model becomes the centre of the detection system, as it can able to find out both the known attacks as well as having the quality to find the unseen attack patterns by analyzing the real-time data. Three steps are involved in the training process. The first step is the loss function in which it will mention the error or divergence between the existence label and the predicted output (The system will classify the action as normal or

intrusion). The basic motive of the training procedure is to reduce the error. The second step is the Graph Sampling, in which it will determine whether the model can able to scale the entire graph at once or not. The third step is the Regularization technique in which it will use dropout and L2 regularization to prevent the over-fitting.

Evaluation: Security has grown to be a major worry for network systems due to the quick expansion of applications and network uses. The self-created system that many IoT devices and CPS rely on is vulnerable to a variety of threats. Sniffers, gateway attacks, denial of service (DoS) attacks, and unauthorized access are all problems with the network layer. The development of extensive, high-dimension computer networks and the Internet of Things coincides with improvements in IDS. However, we assessed the outcomes of the suggested framework in this section. The evaluation can predicted with the accuracy, precision, f1 score and recall in percentage.

CPS attack detection: The trained model will continuously monitor the incoming data and will apply its potential knowledge to determine the attack. This step can provide two sets of detection, either it will be normal or it is malicious attack.

Intrusion alerts or Response Mechanisms: When a system is detected with malicious attack then it will send alert messages to the security team and provides information about the alerted network. In response mechanism, sometimes the system will take immediate action like isolating the affected devices without any user guidance.

Advantages of the Proposed System: Some of the major advantages of this proposed framework when in comparison with the traditional intrusion detection system are,

1. Designing the critical interaction: The GNN are capable to have a communication between the components in CPS. Hence it can able to find out whether the system is attacked or not.
2. Dynamic Action: Since the system can able to handle the time differing data, then it can take quick decision for the real-time data when the system is subjected to any intrusion or network attack.
3. Scalability: Some techniques such as Graph sampling and message passing algorithms enable the GNN to monitor or protect or prevent the system from attack more than once at a time.

IV. RESULTS & DISCUSSIONS

Use The distribution of the protocols [(i.e) the types of protocols availability in the datasets] are represented by the x-axis and the total number of count [(i.e) the minimum number of IP address selected in the datasets] are represented on the y-axis.

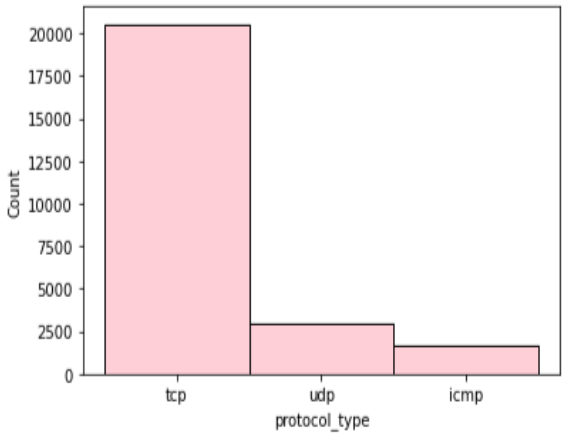

Figure 3: Results of the protocol type

The general comparison of the source bytes (the number bytes sent from the source either the attacker or the legitimate user) and destination bytes (represents the amount of data received by the destination byte). The source byte will represented in the x-axis and the destination byte will represent in the y-axis.

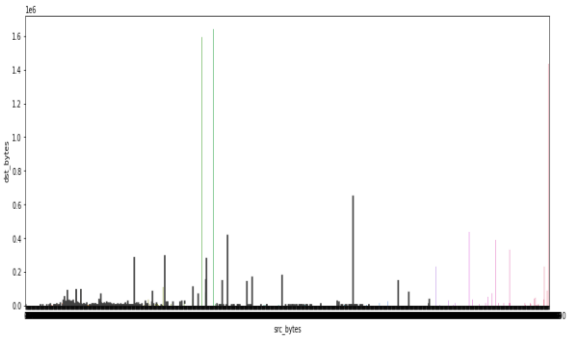

Figure 4: Results of the comparison between source byte and destination byte

The comparison between the normal attack i.e logged in (this may represent the normal or compromised account) represented in x-axis and the brute force attack (High number of failed login will represent the possibility of attack happened in the system) represented in y-axis.

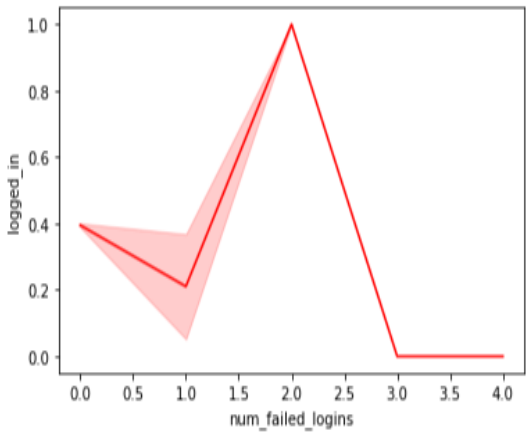

Figure 5: Results of the comparison between logged in and failed login

The source bytes, destination bytes, logged in, failed login are some of the factors present in the datasets to determine the intrusion attack. Remaining factors are represented using the Heat Map.

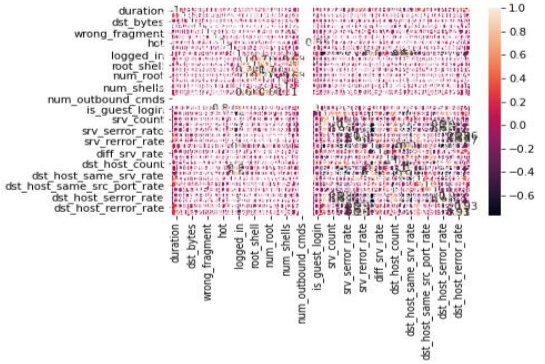

Figure 6: Results of the overall factors

EXPERIMENTAL RESULT

Generally the CNN(Convolutional Neural Network) and GNN (Graph Neural Network) is used for the evaluation purpose. The performance metrics can be evaluated using the accuracy, precision, recall, F1 score and loss.

Metric	CNN Values	GNN Values
F1 Score	0.87	0.92
Accuracy	0.88	0.95
Precision	0.89	0.95
Recall	0.85	0.90
Loss	0.25	0.18
Cohen's Kappa	0.85	0.84

Table 1: Comparison Table for CNN and GNN

The above table represents the comparison between the algorithms among which the accuracy for the CNN is 0.88 and GNN is 0.95. Generally, the increment in the accuracy value shows that the GNN is well trained to predict the intrusion attacks. Similarly, the loss percentage in CNN is 0.25 and GNN is 0.18 The decrement in the loss value represents that the GNN is well trained to predict the intrusion attacks.

V. CONCLUSIONS

In this paper it investigated the usage of Graph Neural Network (GNN) to identify the threat in the Cyber Physical Systems(CPS). For the existing CPS the available traditional methods are often struggle to determine whether the system is attacked or not due its complex nature. This challenges can be easily rectified with the GNN as it can able to understand the intricate relationship between the data. The experiments also determined that the GNN produces high accuracy which are capable to determine the malicious attack in the CPS. This method also ensures the scalability factor as the system can able to predict the more than one attack at a time.

VI. REFERENCES

[1] Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J., "Survey of intrusion detection systems: techniques, datasets and challenges", (2019)

[2] Sahani, N., Zhu, R., Cho, J., & Liu, C., "Machine Learning-based Intrusion Detection for Smart Grid Computing: A Survey", (2023)

[3] Ding, D., Han, Q., Ge, X., & Wang, J., "Secure State Estimation and Control of Cyber-Physical Systems: A Survey", (2020)

[4] Greer, C., Burns, M., Wollman, D., & Griffor, E.," Cyber-physical systems and internet of things", (2019)

[5] Yaacoub, J. A., Salman, O., Noura, H. N., Kaaniche, N., Chehab, A., & Malli, M., "Cyber-physical systems security: Limitations, issues and future trends", (2020)

[6] Friji, H., Olivereau, A., & Sarkiss, M., "Efficient Network Representation for GNN-Based Intrusion Detection", (2023)

[7] Irfan ali kandhro., sultan m., Alanazi kanwal fatima1., mueen uddin., "Detection of real-time malicious intrusions and attacks in iot

empowered cybersecurity infrastructure", (2023)

[8] M. A. Umer., k. N. Junejo., m. T. Jilani., and a. P. Mathur., ''Machine learning for intrusion detection in industrial control systems: applications, challenges, and recommendations'', (2022)

[9] V. Ponnusamy., m. Humayun., n. Z. Jhanjhi., a. Yichiet., and m. F. Almufareh., ''Intrusion detection systems in internet of things and mobile ad-hoc networks", (2022)

[10] M. Sarhan., s. Layeghy., and m. Portmann., ''Towards a standard feature set for network intrusion detection system datasets'', (2021)

[11] Kumar, P., Lin, Y., Bai, G., Paverd, A., Dong, J. S., & Martin, A., "Smart Grid Metering Networks: A Survey on Security, Privacy and Open Research Issues", (2019)

[12] Scalzo, S., & Boissiere, F., "Security Risk Management for the Internet of Things: Technologies and Techniques for IoT Security, Privacy and Data Protection", (2020)

[13] Damopoulos, D., Menesidou, S. A., Kambourakis, G., Papadaki, M., Clarke, N., & Gritzalis, S., "Evaluation of anomaly-based IDS for mobile devices using machine learning classifiers", (2011)

[14] Da Costa, K. A., Papa, J. P., Lisboa, C. O., Munoz, R., & De Albuquerque, V. H. C. "Internet of Things: A survey on machine learning-based intrusion detection approaches", (2019)

[15] Verma, A., & Ranga, V., "Machine Learning Based Intrusion Detection Systems for IoT Applications", (2019)

[16] Catillo, M., Pecchia, A., & Villano, U., "CPS-GUARD: Intrusion detection for cyber-physical systems and IoT devices using outlier-aware deep autoencoders", (2023)

[17] Bitton, R., & Shabtai, A., "A Machine Learning-Based Intrusion Detection System for Securing Remote Desktop Connections to Electronic Flight Bag Servers", (2021)

[18] Sharma, A., Rani, S., Shah, S. H., Sharma, R., Yu, F., & Hassan, M. M., "An Efficient Hybrid Deep Learning Model for Denial-of-Service Detection in Cyber Physical Systems", (2023)