Anomaly-based Intrusion Detection System for IoT Environment using Machine Learning

Arya Patil
School of E&TC Engineering
MIT Academy of Engineering
Pune India

Digvijay Machale
School of E&TC Engineering
MIT Academy of Engineering
Pune India

Dipanshu Goswami
School of E&TC Engineering
MIT Academy of Engineering
Pune India

Parimal Muley
School of E&TC Engineering
MIT Academy of Engineering
Pune India

Prachi Rajarapollu
School of E&TC Engineering
MIT Academy of Engineering
Pune India
prrajarapollu@mitaoe.ac.in

***Abstract*—IoT solutions have the potential to revolutionize our work and daily lives by providing the valuable data and insights. From enhancing the safety of roads, automobiles, and houses to fundamentally improving the way we produce and consume things. Regardless of the IoT system's advantages, some high-profile cyber attacks are preventing many firms from implementing IoT technologies. The current landscape of IoT ecosystem is characterized by complexity. Virtually any industry's equipment and objects can now be interconnected and configured to transmit data to cloud applications and back-end systems through cellular networks. Throughout the entire IoT journey, there is an inherent risk to digital security, with numerous hackers ready to exploit any vulnerabilities in the system. So, the need for security is very much important in IoT. In this paper, IoT system has been built to collect the dataset and perform intrusion to provide details of the data before and after the intrusion. Various machine learning models are used to classify tempered data.**

Keywords—IoT, Intrusion, Machine learning, cyber security, hackers

I. INTRODUCTION

IoT is modern technology consisting of a collaborative network of devices and the cloud, which facilitates communication between themselves. Despite the effectiveness, the security of the system is not well developed. Currently, in the domain of IoT, there are some kinds of intrusion have been seen. The majority of earlier IoT applications had either no security at all or very limited security. This persisted for several days. It took a long time before the value of data was properly understood. Before that time, data was thought to be helpful but not something that should be guarded. When IoT applications became popular, the genuine need for security was felt more than ever before because of financial and personal data. They were aware that data stored on computers is a crucial component of modern life. Hence various security sectors started to acquire popularity. In this project data collection is done comprising of both "Normal Data" and "Attacked Data" to test an intrusion utilizing machine learning. A test-bed consisting IoT based sensor and controller units is designed for the testing as there are no open datasets for analyzing traffic data trends in IoT. Data is recorded in two ways: during neutral traffic that flows normally and during attacks that are launched by malicious devices. A test bed is constructed in the first step to replicate an IoT-based system. The second stage involves creating a system that launch cyber attacks against the network. The third stage comprises of machine learning algorithms that are created to identify and categorize network intrusions, and the effectiveness of the filters is evaluated using approved metrics. It will classify attacked data from normal data.

II. Literature Survey

In this paper author has focused on the cyber security related to false detection while using internet globally. How cyber security becomes more important during the usage of global Security Operations Center (SOC) environments. Various machine learning algorithms used to compare the result [01]. As per the author machine learning is the best solution to detect false positive rate of malicious in security operation center. The paper implemented the system in such a manner that, both the groups having the knowledge of machine learning in depth with less information of cyber security functioning, and the people with experts in cyber security with less information related to machine learning algorithms can work and efficiently prepare the system as per requirements. The Symantec SOC production environment, which helps in system implementation right from label selection, selection of engineering features, data collection etc. [02]. The system implemented is focusing on financial transactions and security parameters related to digital transactions. As the digital transactions are increasing day by day security is also is on stake. System is implemented in user friendly manner so that the user can modify and design it as per the requirement [03]. As per the author, financial transactions and it's security is very important, while artificial intelligence is more suitable solution for identification of unwanted threats and avoidance of malicious attacks [04]. The major concern of the author in this paper is the security of industry 4.0. Due to the drastic growth in the industrial sector security parameters are also becomes unavoidable. Various machine learning algorithms are used for detection of cyber and networks attack. Comparative analysis is also carried out in the paper [05].

III. DESIGN & IMPLEMENTATION

The DHT11 sensor and Node MCU module is connected to build IoT Platform for testing. The three pins on the DHT11 sensor are connected to the Node MCU's VCC, GND, and D1 pins, to gather the data obtained from the DHT11 sensor. Created a channel on the ThinkSpeak

979-8-3503-1587-5/23/$31.00 ©2023 IEEE

Cloud server to store the data and wireless router is connected to store the values of the sensor. The "ESP8266WiFi.h" library is used for establishing the connection between Node MCU and ThinkSpeak Cloud Server through wireless router. The system is created with the Kali Linux platform to perform man in the middle attack by using Burp Suite testing tool. After performing Man in the middle attack duplicate values are inserted and the collected data labeled as attack data. The two different dataset which is 'normal data 'and 'attacked data' are classified by using different machine learning techniques such as SVM, Isolation Forest, Naïve Bayes and KNN algorithms. Compared the accuracy of all algorithm by plotting confusion matrix to determine which algorithms is the best choice for intrusion detection. Figure 1 shows the block diagram of the system implemented.

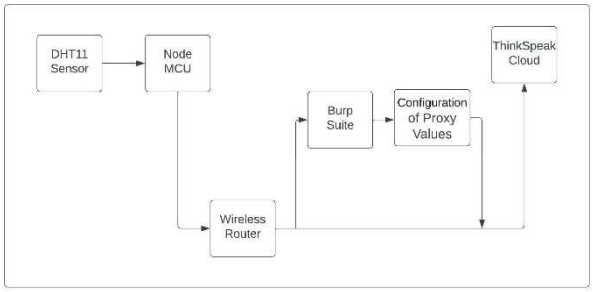

Figure 1: Block Diagram for Implemented System

A. Building IOT Test Bed for Data Collection (Phase 1)

The DHT11 sensor and Node MCU will give us the necessary test environment. ThinkSpeak offers channels that save all of the information transmitted by the system. Many fields on each channel are used to store the various packets of data that the Node MCU controller sends. To track the information collected by the sensor, a channel with the two fields namely 'Temperature' and 'Humidity' are constructed. An installation of ESP8266 and DTH sensor libraries will be required to use the DHT11 temperature sensor and ESP8266 Wi-Fi module. On the ThinkSpeak server, the channel that will receive transmissions of sensor data is provided with unique API keys. With the aid of server-side API calls, values captured by sensors are sent to the ThinkSpeak cloud platform for storage and analysis.

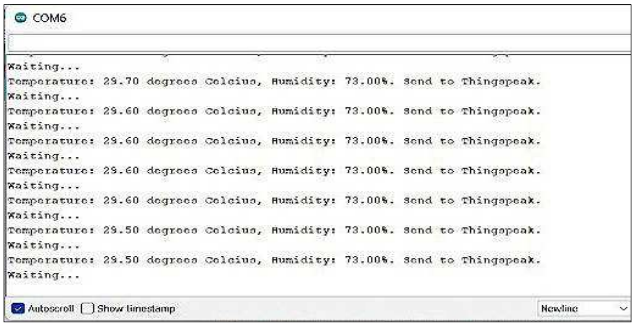

Figure 2: Data Collection using ESP8266 and DHT11 sensor

B. Data Collection from the ThinkSpeak Cloud

The values which are sent from NodeMCU are uploaded to the Cloud. Two different fields have been created for the sensor values. One field is for the Temperature Value and the Second field is for the Humidity value. In ThinkSpeak Server Webpage, data is visualized with the time stamp. The recorded readings of the sensors are exported in the .csv file. And the dataset is used for machine learning model building. Data collection is done on large scale as shown in figure 2. Graphical representation of data collected can be observed in figure 3.

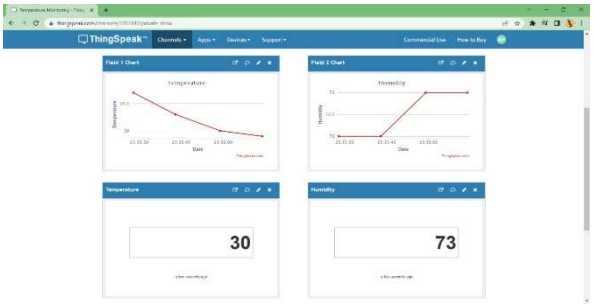

Figure 3: Data Collection from the ThinkSpeak Cloud

C. Man In the Middle Attack for Intruding the Server (Phase 2)

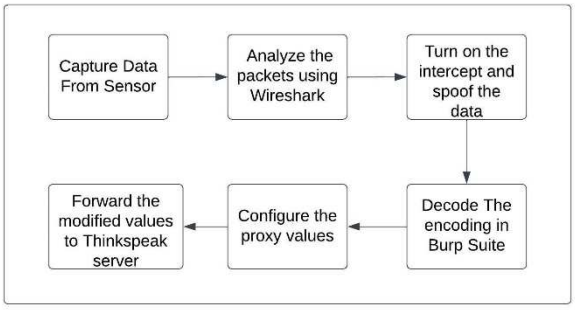

Figure 4: Attack Phase Block Diagram

To create intrusion between the loT network platform, an adversarial system has been developed. Kali Linux is a platform which offers a different network penetration testing tools like Wireshark, Burp Suite, Ettercap, etc. Kali Linux can also operate by using Oracle VM VirtualBox. Wireshark, a widely recognized and commonly used network protocol analyzer, provides users with the ability to capture and analyze real-time network traffic. Wireshark is used to perform a preliminary analysis on data packets containing sensor data before they are forwarded to the ThinkSpeak server. Burp Suite, an extensively utilized web application security testing tool created by PortSwigger, empowers security professionals to detect vulnerabilities and potential security risks within web applications. Interception is created between the Router and ThinkSpeak Cloud, so every request which is sent from Node MCU through wireless router can be seen on Burp Suite. For creating the intrusion some proxy values are added and the requests have been forwarded. Figure 4 shows the attack phase block diagram. Two different datasets are created.

The one which labeled as 'Normal Data' in which no intrusion has created so all values are directly pass through router towards the cloud. In second Dataset the intrusion has been created and some proxy values are added and labeled as 'Attacked Data' as shown in figure 5.

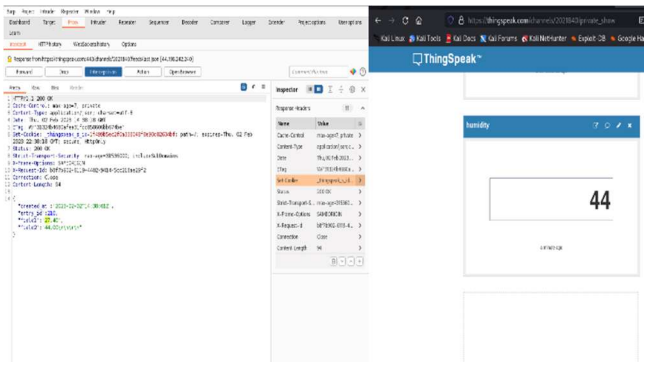

Figure 5: Proxy Values added to the System

D. Building Machine Learning Models to classify the data (Phase 3)

In this section, machine learning algorithms are implemented for the detection and classification of attacks. SVM, Naïve Bayes, KNN and Isolation Forest these are the four algorithms are used for detection and classification for intruded data. Support Vector Machine (SVM) is a supervised learning algorithm widely utilized in both classification and regression tasks. The main objective of SVM is to find the optimal hyperplane that separates different classes or groups of data points in the feature space. The Naive Bayes algorithm is a machine learning technique that leverages Bayes' theorem for classification tasks. It is a probabilistic model that is used for classification and is particularly useful when dealing with large datasets. Naive Bayes is a popular algorithm for classification tasks because it is simple, fast, and efficient. The Naive Bayes algorithm operates by estimating the probabilities of different class labels given the observed features of a new instance. The class label with the highest probability is then assigned to the new instance. The core concept of the k-nearest neighbors (KNN) algorithm revolves around classifying a new data point by examining the class labels assigned to its closest neighbors in the feature space. The Isolation Forest is an ensemble learning method having basic idea to isolate data points by constructing a forest of decision trees, where each tree is grown by randomly selecting a feature and a split point. The algorithm then calculates the number of splits required to isolate each data point. Anomalies, which are points that require fewer splits to isolate, are identified as potential outliers. All four classifiers are effective in creating binary classifiers, which can be used to distinguish between attack and normal classes during classification.

IV. RESULT

A. Implementation

Machine learning (ML) anomaly-based intrusion detection is a technique for finding out-of-the-ordinary patterns or behaviors in a system or network that might point to an intrusion or attack. ML algorithms are used to discover variations from these patterns, which may be signs of malicious activity, after learning the typical patterns of system behavior.

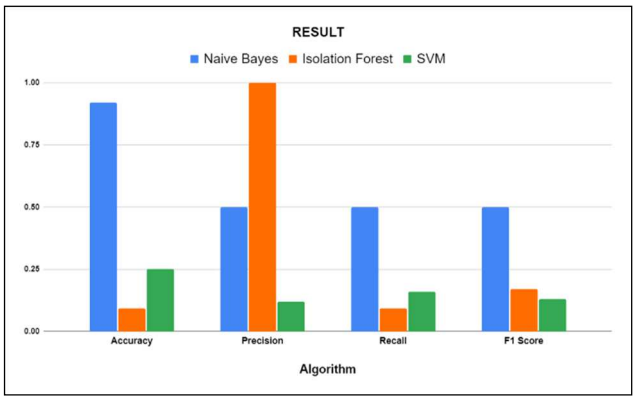

Figure 6: Comparison of Algorithmic Accuracy

B. Naive Bayes

A machine learning method called Naive Bayes is frequently employed in intrusion detection systems to categories network traffic or system behavior as legitimate or malicious. It takes as its premise that the features used for classification are conditionally independent and is based on the Bayes theorem. The algorithm determines the likelihood that an instance belongs to a specific class (normal or malicious) based on the likelihoods that each feature occurs in each class. Naive Bayes is a good choice for real-time intrusion detection since it is effective and affordable computationally.

C. Isolation Forest

To find abnormalities or outliers in network traffic or system behavior, intrusion detection systems frequently use the machine learning algorithm isolation forest. It works by building a binary tree structure with the leaf nodes representing isolated cases or anomalies and the inside nodes representing a splitting condition on a selected feature. The technique is based on the idea that anomalies are often few and distinctive, making it easier to identify them from normal instances. The number of splits necessary to isolate an instance is used to calculate the anomalous score, with examples requiring fewer splits receiving higher scores.

D. SVM

A common machine learning approach used in intrusion detection systems is called Support Vector Machines (SVM). SVM divides various groups, such as legitimate and malicious network traffic by the process of finding the optimal hyperplane in support vector machines (SVM) involves mapping the input data into a higher-dimensional feature space. When dealing with high-dimensional data, SVMs are excellent in classifying instances according to how close they are to the decision border. SVMs may learn the patterns and traits of legitimate and harmful operations in the context of intrusion detection, helping them to

precisely categories new occurrences. SVMs offer strong and reliable intrusion detection capabilities by maximizing the margin between classes, which makes them an effective tool for defending networks from threats. Table 1 gives the complete performance measures of various algorithms implemented.

Table 1: Performance Measures of Different ML Algorithms

Algorithm	Accuracy	Precision	Recall	F1 Score
Naive Bayes	0.92	0.5	0.5	0.5
Isolation Forest	0.093	1	0.093	0.17
SVM	0.25	0.12	0.16	0.13

V. CONCLUSION

IoT networks need to be protected from attacks because they are very susceptible to them, so solutions must be developed and put to the test. An IoT-based platform is developed for this project and used as a proving ground for understanding and carrying out IoT assaults on the network. The use of machine learning techniques for intrusion detection, such as Naive Bayes, Support Vector Machines (SVM), and Isolation Forest, provides efficient ways to find and fix security flaws.

A probabilistic technique called Naive Bayes determines, based on the existence of particular traits, the likelihood that an instance belongs to a given class. It can effectively handle vast feature spaces and assumes that each feature is independent. Despite being computationally simple, Naive Bayes may have trouble with complex correlations and interactions between features. On the other hand, SVM identifies the best hyperplane to divide classes by mapping input data into a high-dimensional feature space. Instances that are close to the decision boundary can be successfully classified using this method, which excels at handling high-dimensional data. SVMs maximize the margin between classes to provide strong intrusion detection capabilities. An unsupervised learning system called Isolation Forest isolates instances in a binary tree structure to identify anomalies. Shorter pathways are seen as anomalies since 25 the metric counts the splits needed to isolate an instance. Isolation Forest is a good choice for outlier discovery in intrusion detection systems since it is particularly good at finding uncommon and previously undiscovered attacks.

These algorithms each have advantages and disadvantages. Though straightforward and computationally effective, Naive Bayes may have trouble handling complex interactions. SVMs provide reliable classification but can be expensive computationally for large datasets. Although successful in finding abnormalities, Isolation Forest occasionally generates false positives. The specific needs, dataset features, and trade-offs between detection accuracy, computational complexity, and resource limitations all play a role in determining which algorithm should be used. The effectiveness and dependability of intrusion detection systems can be further improved by combining these algorithms or ensemble techniques.

REFERENCES

[1] S. Kumar, B. P. Singh and V. Kumar, "A Semantic Machine Learning Algorithm for Cyber Threat Detection and Monitoring Security," *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, Greater Noida, India, 2021, pp. 1963-1967

[2] C. Feng, S. Wu and N. Liu, "A user-centric machine learning framework for cyber security operations center," *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, Beijing, China, 2017, pp. 173-175

[3] D. Kumar and K. P. Kumar, "Artificial Intelligence based Cyber Security Threats Identification in Financial Institutions Using Machine Learning Approach," *2023 2nd International Conference for Innovation in Technology (INOCON)*, Bangalore, India, 2023, pp. 1-6

[4] H. M. Farooq and N. M. Otaibi, "Optimal Machine Learning Algorithms for Cyber Threat Detection," *2018 UKSim-AMSS 20th International Conference on Computer Modelling and Simulation (UKSim)*, Cambridge, UK, 2018, pp. 32-37

[5] F. S. Cebeloglu and M. Karakose, "Comparative Analysis of Cyber Security Approaches Using Machine Learning in Industry 4.0," *2020 IEEE International Symposium on Systems Engineering (ISSE)*, Vienna, Austria, 2020, pp. 1-5

[6] T. M. Ghazal, M. K. Hasan, R. A. Zitar, N. A. Al-Dmour, W. T. Al-Sit and S. Islam, "Cybers Security Analysis and Measurement Tools Using Machine Learning Approach," *2022 1st International Conference on AI in Cybersecurity (ICAIC)*, Victoria, TX, USA, 2022, pp. 1-4

[7] S. Dhir and Y. Kumar, "Study of Machine and Deep Learning Classifications in Cyber Physical System," *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, 2020, pp. 333-338

[8] A. O. Al-Ansari and T. M. Alsubait, "Predicting Cyber Threats Using Machine Learning for Improving Cyber Supply Chain Security," *2022 Fifth National Conference of Saudi Computers Colleges (NCCC)*, Makkah, Saudi Arabia, 2022, pp. 123-130

[9] S. Singhal, R. Srivastava, R. Shyam and D. Mangal, "Supervised Machine Learning for Cloud Security," *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, India, 2023, pp. 1-5

[10] M. Vargheese, G. Nallasivan, D. D. N. Ponkumar, N. Ponnithish, P. K. Devi and M. Arun, "Machine Learning for Enhanced Cyber Security," *2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, 2023, pp. 709-713