International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

Robust and Reliable Security Approach for IoMT: Detection of DoS and Delay Attacks through a High-Accuracy Machine Learning Model

Abdullah Ali Jawad Al-Abadi[1], Mbarka Belhaj Mohamed[2], Ahmed Fakhfakh[3]
[1]Laboratory of signals, systems, artificial intelligence and networks (SM@RTS), Digital Research Center of Sfax (CRNS), National School of Engineers of Sfax (ENIS)
University of Sfax
Sfax, Tunisia
abdullah.jawad.1980@gmail.com
[2]Laboratory of signals, systems, artificial intelligence and networks (SM@RTS), Digital Research Center of Sfax (CRNS), National School of Engineers of Gabes (ENIG)
University of Sfax
Gabes, Tunisia
mbarkaenig@gmail.com
[3]Laboratory of signals, systems, artificial intelligence and networks (SM@RTS), Digital Research Center of Sfax (CRNS), National School of Engineers of Sfax (ENIS)
University of Sfax
Sfax, Tunisia
ahmed.fakhfakh@enetcom.usf.tn

Abstract—Internet of Medical Things (IoMT) refers to the network of medical devices and healthcare systems that are connected to the internet. However, this connectivity also makes IoMT vulnerable to cyberattacks such as DoS and Delay attacks , posing risks to patient safety, data security, and public trust. Early detection of these attacks is crucial to prevent harm to patients and system malfunctions. In this paper, we address the detection and mitigation of DoS and Delay attacks in the IoMT using machine learning techniques. To achieve this objective, we constructed an IoMT network scenario using Omnet++ and recorded network traffic data. Subsequently, we utilized this data to train a set of common machine learning algorithms. Additionally, we proposed an Enhanced Random Forest Classifier for Achieving the Best Execution Time (ERF-ABE), which aims to achieve high accuracy and sensitivity as well as low execution time for detecting these types of attacks in IoMT networks. This classifier combines the strengths of random forests with optimization techniques to enhance performance. Based on the results, the execution time has been reduced by implementing ERF-ABE, while maintaining high levels of accuracy and sensitivity.

Keywords- IoMT; DDoS attack; Delay Attack; Logistic Regression; Random Forest; Decision Tress; Stochastic Gradient Distance; Naive Bayes.

I. INTRODUCTION

A cyberattack in the IoMT is a malicious attempt to compromise the security and integrity of medical devices and systems that are connected to the internet. IoMT devices include a wide range of medical equipment and wearable devices, such as pacemakers, insulin pumps, blood glucose monitors, and fitness trackers [1].

A cyberattack in IoMT can have severe consequences, including patient harm, loss of sensitive data, financial losses, and damage to public trust. Therefore, it is critical to implement strong cybersecurity measures to protect IoMT devices and systems from potential threats.

A DoS attack includes flooding a system or network with traffic or requests, causing the system to slow down or stop working entirely. This type of attack can be dangerous in IoMT if it causes a medical device or system to malfunction, potentially putting the patient's health at risk. For example, if a pacemaker is targeted by a DoS attack, it could stop working altogether, which could be life threatening for the patient.

A delay attack involves altering the timing of data transmissions between IoMT devices and networks. This can lead to delays in critical information, which could be detrimental to patient care. For example, a delay in transmitting data from a blood glucose monitor to a healthcare provider could result in a patient's blood sugar levels going unmonitored for an extended period, potentially causing harm.

Detecting and mitigating DDoS (Distributed Denial of Service) and Delay Attacks is crucial for ensuring the availability and reliability of online services. Machine learning

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

algorithms have emerged as a promising solution for detecting such attacks by analyzing traffic patterns and identifying anomalies that may indicate an ongoing attack. Here are some reasons why using machine learning in detecting DDoS and Delay attacks are important:

Early detection: Machine learning can analyze traffic patterns and identify anomalies that may indicate an ongoing attack. Early detection is crucial in preventing an attack from causing significant damage [2].

Improved accuracy: Machine-learning algorithms can analyze large volumes of traffic data and identify patterns that can probably be missed by human analysts. This can help to improve the accuracy of detection and reduce false positives.

Adaptability: Machine-learning models can be trained on large datasets of historical traffic data and can adapt to new attack patterns as they emerge. This can make them more effective in detecting and mitigating attacks.

Automation: Machine learning can automate the detection and mitigation of DDoS and Delay attacks, allowing security teams to respond quickly and effectively [3]. This can reduce the impact of an attack and prevent data loss or service disruptions.

Machine learning can provide a valuable tool in the fight against DDoS and Delay attacks. By analyzing large volumes of data, identifying patterns, and adapting to new attack methods, machine learning can help to detect and prevent attacks before they cause significant damage.

II. CYBER ATTACKS AND SECURITY MEASURES

A. CYBER ATTACKS

Cyber-attacks are planned efforts to take advantage of flaws in computer networks, devices, or systems with the intention of inflicting damage or obtaining unauthorized access to confidential data. Numerous sorts of cyberattacks are possible, including [4]:

Malware attacks: Software designed to harm networks or computer systems is known as malware. Malware may take many forms, including but not limited to viruses, worms, Trojan horses, ransomware, and spyware.

Phishing attacks: Phishing is a kind of social engineering where perpetrators send messages or emails that seem to come from a reliable source in an effort to dupe recipients into credit card details, reveal personal information, or other sensitive information.

Denial-of-service (DoS) attacks: A DoS attack aims to overload a system or network with traffic or requests, forcing it to crash or stop functioning.

Man-in-the-middle (MITM) attacks: MITM involve intercepting communications between two parties in order to steal or alter data.

A Delay attack: In a delay attack, the adversary purposefully slows the transmission of data packets or messages over a network. By taking advantage of flaws in the network architecture or by using strategies like packet buffering and queue manipulation, the attacker may do this.

Cyber-attacks can have a wide range of impacts, including financial losses, reputational damage, and even physical harm in some cases. Use strong passwords, update software often, and exercise caution when opening email attachments or clicking on links to protect yourself and your systems against cyber-attacks.

B. Security Requirements

Security requirements are a set of standards, procedures, and guidelines that organizations and individuals must follow to ensure the confidentiality, integrity, and availability of their information, systems, and assets [5]. Security requirements help to identify and mitigate potential security risks, threats, and vulnerabilities.

Security requirements can be broadly classified into three categories:

- Physical security requirements: These requirements focus on protecting physical assets such as buildings, servers, and other hardware from theft, damage, or unauthorized access. Examples of physical security requirements include access controls, surveillance cameras, locks, and alarms.
- Technical security requirements: These requirements focus on protecting software, systems, and networks from cyber attacks, malware, and other security threats. Examples of technical security requirements include firewalls, intrusion detection systems, encryption, and strong authentication mechanisms.
- Administrative security requirements: These requirements focus on creating policies, procedures, and guidelines to ensure the proper handling and use of sensitive data and assets. Examples of administrative security requirements include background checks, security awareness training, incident response plans, and disaster recovery procedures.

Protecting sensitive information, avoiding data breaches, and upholding stakeholder and consumer confidence all depend on adherence to security rules. It is critical to regularly assess and update security standards to account for new vulnerabilities and threats.

The three main tenets of information security are confidentiality, integrity, and availability (CIA), which are sometimes shortened as CIA. The design, implementation, and administration of security controls and procedures are based on these guiding principles in order to safeguard sensitive data and information systems against illegal access, alteration, or destruction.

- Confidentiality: Confidentiality is the principle of ensuring that information is not disclosed to

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

unauthorized individuals or entities. This means that sensitive information is kept secret and only accessible to authorized users who have a need-to-know basis. Examples of confidentiality measures include encryption, access controls, and data classification.

- Integrity: Integrity is an idea where the data is correct and full, and it remained unchanged or tampered with authorization. This implies that modifications to data should be monitored and audited [5], and data should not be changed or deleted without sufficient authorization. Checksums, digital signatures, and access constraints are a few examples of integrity controls. Measures like checksums, digital signatures, and access restrictions are often used for assuring data integrity.
- Availability: The idea of availability is to make sure that when required, authorized users may access and utilize information and services. This calls for the appropriate operation and functioning of systems as well as the timely availability and retrieval of data. Disaster recovery plans, redundancy, and backup and recovery systems are a few examples of availability measures.

Together, confidentiality, integrity, and availability form the foundation of information security and are critical to protecting sensitive information and systems from cyber attacks, data breaches, and other security threats [6].

C. Denial of Service and Distributed Denial of Service

By flooding a network, website, or service with a lot of requests or traffic, DoS (Denial of Service) and DDoS (Distributed Denial of Service) both are cyberattacks' forms that aim to prevent it from being available. The attack's origin is the primary distinction between DoS and DDoS.

A single computer or device is used in a DoS attack to bombard a target with traffic or requests, making it unusable or crashing. A single attacker or a small group of attackers often conducts this kind of assault utilizing amplification tactics or tools like botnets. Jamming attacks [7], Ping floods, SYN floods, and HTTP floods are a few examples of DoS attacks.

In a DDoS attack, multiple computers or devices are used to flood a target with traffic or requests from different locations or sources, making it more difficult to defend against the attack. A larger group of attackers using compromised computers or devices that are part of a botnet often carries out this type of attack. Examples of DDoS attacks include UDP floods, DNS amplification attacks, and HTTP POST floods [8].

Both DoS and DDoS attacks can have serious consequences for the targeted organization, including downtime, lost revenue, reputational damage, and even legal liabilities. To mitigate the risk of DDoS and DDoS attacks, organizations can implement various security measures, such as traffic filtering, rate limiting, intrusion detection and prevention, and distributed server architectures.

D. Delay Attack

In the context of cybersecurity, a delay attack is a type of attack where an adversary intentionally delays the delivery of data packets or messages in a network. The attacker may do this by employing methods like packet buffering, queue manipulation, or routing manipulation, or by taking advantage of flaws in the network architecture. The objective of a delay attack can vary depending on the attacker's motivation, but it often involves disrupting the normal operation of the network, causing congestion, or impeding the delivery of critical data [9].

Delay attacks can have severe consequences in various applications, such as financial transactions, real-time monitoring systems, healthcare systems or industrial control systems, where delays in data delivery can lead to significant safety impacts. Detecting and mitigating delay attacks often requires specialized techniques, such as traffic analysis, anomaly detection, or secure routing protocols. In addition, implementing network redundancy and diversity can reduce the impact of delay attacks by providing alternate communication paths and ensuring the availability of critical services.

E. Machine Learning

A collection of mathematical theories and methods known as machine learning algorithms enable computers to learn from experience and advance without explicit programming 9. These algorithms make up the foundation of artificial intelligence (AI) and are utilized in a variety of tasks [10], including fraud detection, natural language processing, image identification, and image recognition.

Here are some of the widely used machine learning algorithms:

- Logistic regression: A binary outcome variable can be predicted using the statistical technique of logistic regression using one or more predictor variables. It is often used in projects like fraud detection and churn prediction.
- Decision trees: A machine learning technique known as a "decision tree" models choices and their outcomes using a tree-like structure. They are often used for activities like product suggestion and client segmentation.
- Random forests: Random forests are an ensemble learning algorithm that combines multiple decision trees to improve predictive accuracy and reduce overfitting. They are commonly used for tasks such as image classification and credit risk assessment.
- Naive Bayes: A probabilistic method called Naive Bayes is based on the Bayes theorem. It is often used

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

for applications like spam filtering and text categorization. The algorithm is referred to as "naive" since it presumes that the characteristics are independent of one another. Naive Bayes is often successful and economical despite this oversimplifying assumption, particularly for high-dimensional datasets.

- Stochastic Gradient Descent (SGD): A popular sort of optimization strategy for training machine-learning models is stochastic gradient descent. For big datasets, when the conventional batch gradient descent approach becomes computationally costly, it is advantageous. SGD updates the model's parameters progressively rather than all at once, using tiny random chunks of the training data.
- K-Nearest Neighbors (KNN): A popular non-parametric technique for classification and regression applications is K-Nearest Neighbors. A KNN predicts the output value based on the labels of the k neighbors that are the k closest data points to the input data point in the training set. K's value is a hyperparameters that may be adjusted to provide the best results.

These are just a few examples of the many machine-learning algorithms that exist. The choice of algorithm depends on the specific problem, the available data, and the desired outcomes.

Machine learning is increasingly being used in the detection of cyber attacks. The ability of machine learning algorithms to learn patterns and anomalies in large volumes of data makes them well-suited for this task.

Here are some examples of how machine learning can be used in the detection of cyber attacks:

- Intrusion detection: Machine learning can be used for detecting anomalies in network traffic, which may indicate the presence of an intruder. This can be done by training a model on a large dataset of normal network traffic and then using it to detect any deviations from this pattern [11].
- Malware detection: Machine learning can be used to detect malware by analyzing features such as file size, file type, and file behavior. A machine-learning model can be trained on a dataset of known malware to identify new malware that has not been previously seen.
- Phishing detection: Machine learning can be used to detect phishing emails by analyzing the email content, sender information, and other metadata. A machine-learning model can be trained on a dataset of known phishing emails to identify new ones [12].
- Fraud detection: Machine learning may help detect fraudulent purchases by looking for patterns and outliers in transaction data. A machine-learning model may be trained on a dataset of previously identified fraudulent transactions to help identify future ones.

Machine learning may also be used with additional security measures like firewalls and security information and event management (SIEM) systems to provide a more complete protection against cyberattacks [13]. Machine learning models must be continuously reviewed and updated to maintain their efficacy, but it is also important to remember that they might be open to attacks.

F. Principal Component Analysis (PCA)

In very big data sets, the principal component analysis (PCA) approach is used for minimizing the number of dimensions. For achieving this goal, it takes a large collection of variables and cuts them down to a smaller set while keeping much of the information that was there before.

Reduced accuracy is inevitable when a dataset's number of variables is decreased. The idea behind dimensionality reduction is for trade off some precision for simplicity, since smaller data sets are simpler for examining and interpreting and as machine-learning algorithms can analyze data much faster and easily with fewer additional variables to maintain.

The five stages of principal component analysis are as follows

1) Step 1: Standardization

The continuous initial variables are standardized in this step to ensure that each one contributes equally to the study. PCA must be standardized first since it is sensitive to initial variable variances. In other words, If the ranges of the starting variables are quite different from one another, the one with the larger range (say, between 0 and 100) will dominate the one with the smaller range (say, between 0 and 1), leading to biased results. Data scaling can thus address this problem. In mathematical terms, this is accomplished by taking each variable's standard deviation and subtracting the result from the mean.

2) Step 2: Calculation of the Covariance Matrix.

It establishes if there is a connection between the variables' departures from the mean in the input data set. mainly because redundant information might be present in strongly linked variables. These relationships are seen in the covariance matrix. All potential pairings of starting variables are represented by entries in the covariance matrix, which is a p*p symmetric matrix.

3) Step 3: Determine The Principal Components by computing the Eigenvalues and Eigenvectors of the Covariance Matrix

To determine the main components of the data, the eigenvectors and eigenvalues of the covariance matrix must be determined. Let's begin by defining major components. Principal components are produced by linear combinations or blends of the initial variables. These combinations decouple the main components from the beginning variables and pack a lot of data from the early variables into the primary components.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

10 dimensions of data provide 10 main components. PCA continues to aim to place the majority of the information in the 1st component, then the majority in the 2nd component, and so on. With this basic component structure, dimensionality is decreased without information being lost. Low-information components should be eliminated, and the remaining parts should be used as variables. The initial variables are combined linearly to form the fundamental components, which are less comprehensible and nonsensical. In terms of geometry, fundamental components are the lines that gather the most information and account for the greatest variation. A line disperses more data points the more variation it contains. Line dispersion increases information. Principle components may be thought of as additional dimensions that make it easier to distinguish between observations.

4) Step 4: Feature Vector subsection

As it is seen previously, we may find the key components in order of significance through calculation the eigenvectors and ranking them by eigenvalues in decreasing order. In order to create a matrix of vectors known as feature vectors, we must determine whether to keep all of these components or remove those with low eigenvalues. The eigenvectors of the components we keep are columns in a matrix that makes up the feature vector. The first stage in dimensionality reduction is to remove all except p eigenvectors (components) from the n-dimensional space. There will only be p dimensions in the finished data collection.

5) Step 5: Recast the Data along the Principal Components Axes

Recasting the data along the principal component axes is step five in the procedure.

Without altering the data in any way other than normalization, we choose the primary components and create the feature vector. The initial data set (i.e., the initial variables) always stay in relation to the original axes. In the last step, the data are reoriented from the original axis to the primary components using the feature vector created from the covariance matrix's eigenvectors. To achieve this, we multiply the original data sets with the feature vector's transpose.

III. RELATED WORK

To create network traffic and monitor network metrics like throughput and end-to-end latency, there are many different frameworks available.

For switch testing, researchers [14] suggested a traffic generation tool. The hardware and software pieces that make up the traffic generator are both present. The program produces setups and settings based on the user-selected traffic model. As the software module specifies, the hardware module creates the packets and sends them to the network interface module. The inventors [15] presented a scriptable traffic generator made up of software and hardware elements. The hardware module was built to manage the packet rate and delay, whilst the software module was developed for packet settings.

A traffic creation architecture for evaluating the deep packet inspection (DPI) tools was suggested by the authors [16]. The suggested system simulates user behavior to produce network traffic. To test the DPI tools afterwards, they collected the real-time data, analyzed it, and retrieved the normal user behavior. Researchers [17] created a hardware Internet of Things device that can produce IoT traffic. Depending on the data amount and interval duration, the hardware may provide many traffic flows at once. But layer 2 traffic is the only kind that this gadget produces.

Authors [18] integrated big data with machine type communication (MTC) traffic models to create a framework for a traffic generator. The approach was suggested to assess how well mobile networks function. Authors [19] created a program that not only produces network traffic but also assesses network performance and facilitates switch-level functional testing. To evaluate the switch network's functionality, the tool may be used to create various test scenarios and analyze the results.

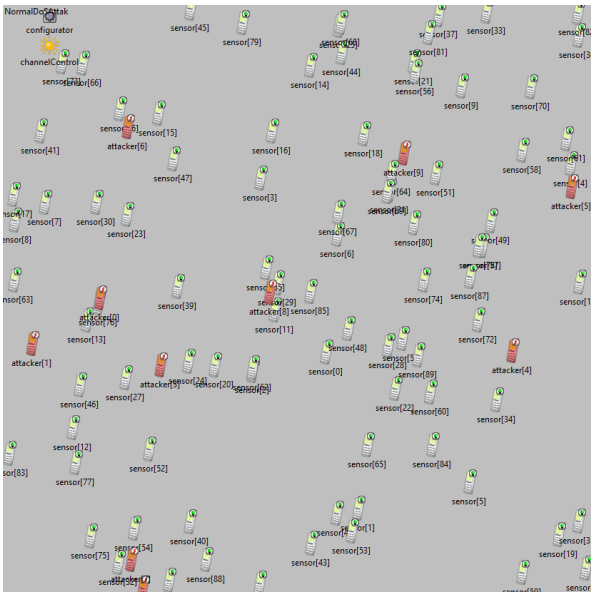

Figure 1. The proposed IoMT network.

Authors [20] suggested a traffic generator that can produce both network layer and transport layer traffic. TCP and UDP protocols are supported by the traffic generator. To create traffic, a user must provide the content, data size, duration, flow attributes, and traffic model.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

The use of Wireless Body Sensor Networks (WBSN) and Internet of Things (IoT) technology for continuous healthcare monitoring is suggested by researchers [21]. To allow illness identification and diagnosis, the major goal is to gather and evaluate physiological data, such as ElectroCardioGram (ECG), ElectroMyoGram (EMG), and Blood Pressure (BP). Data collection, data preparation, and data storage/diagnosis are the three steps of the suggested methodology. Algorithms for supervised machine learning are used for categorization in order to get accurate results. Random Forest (RF), Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), k-Nearest Neighbors (KNN), Gradient Boost (GB), and Naive Bayes (NB) are just a few of the techniques that the researchers compare. The findings demonstrate that RF

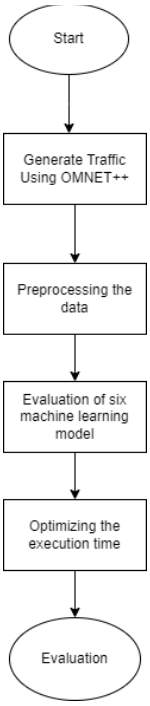

Figure 2. The methodology.

surpasses the other algorithms for the combined ECG, EMG, and BP signals in terms of accuracy (97%) and sensitivity (92%).

IV. MATERIALS AND METHODS

The procedure for this research is outlined in Fig. 1. Initially, we used the Omnet++ simulator to generate network traffic scenarios, including both normal and attack traffic. Network parameters were recorded for each traffic, and the resulting dataset was configured to be used for training different machine learning models. The performance of the developed models was then evaluated in terms of accuracy, sensitivity, and execution time.

A. Traffic Generation

An IoMT network is simulated using OMNET++ 4.3 as simulation software and the Neta 1.1 framework in addition to the INET 3.3 framework as shown in Fig. 2. This was done in order to gather the parameters of this network to generate a dataset. We added 100 IoT nodes to the network. Each nodes contains two types of healthcare monitoring sensors (EEG and ECG), and those nodes distributed the message among themselves to reach the gateway. The parameters of the Network are described in Table I.

TABLE I. THE NETWORK PARAMETERS

Parameter	Value
Network Simulator	Omnet++ 4.3
Framework	INET 3.3
Network Attacks	Neta 1.1
Number of Nodes	2,4,6,8
Area	3000 * 3000 m*m
Simulation Time	300 seconds
Node Mobility	Random Mobility
Packet Size	512 B
Source application	TCP session
Destination application	TCP sink
Number of repetition	AODV
Transmission Protocol	TCP Session APP
Attack Types	DDoS, Delay Attack
Packet Size	512 Bytes

We simulated the network for 300 seconds and recorded the parameters that represent the normal state of the designed network.

Subsequently, we introduced 2, 4, 6, and 8 attack nodes into

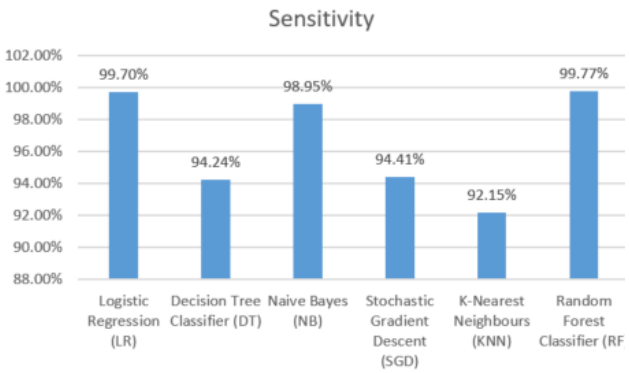

Figure 3. Sensitivity of the six algorithms.

the system, with the attackers implementing either DDoS attacks

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

or delaying the transmission of data. We recorded the parameters that represented the state of the network under attack.

To prepare the dataset for machine learning, we assigned a label to each record indicating whether it was in a normal (0) or attack (1) state. Then we shuffled the dataset to reduce variance and prevent overfitting. Finally, we splatted the dataset into testing and training sets, with 20% of the data allocated for testing set and 80\% for training set.

B. Experiments

1) First Experiment

In this experiment, we trained the most common machine learning algorithms (Logistic Regression (LR), Decision Tree Classifier (DT), Naive Bayes (NB), Stochastic Gradient Descent (SGD), K-Nearest Neighbors (KNN), K-Nearest Neighbors (KNN), Random Forest Classifier (RF)) by using our data that we got it by simulating the attacks on our system and we measured the accuracy, sensitivity and execution time of each of them as shown in Table II.

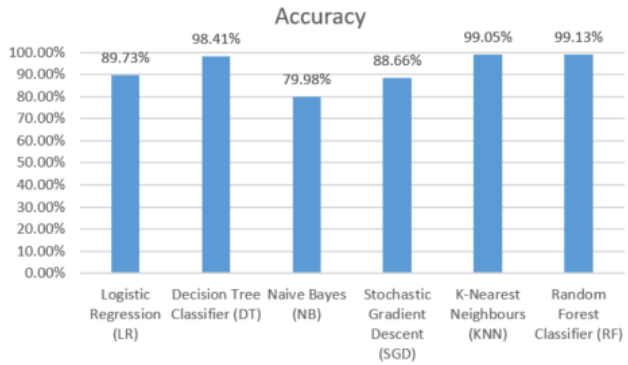

Figure 5. Accuracy of the six algorithms.

The time it takes a computer to finish training or inferring a particular model is called its "execution time." How effectively a model or test can accurately identify the target class or condition is referred to as its accuracy. However, sensitivity evaluates how effectively a model or test can pick out the "positives" from a given collection of data.

Accuracy and Sensitivity are expressed in (1)-(2).

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP+FN}} * 100 \quad (1)$$

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TF+FT+ FN}} * 100 \quad (2)$$

TABLE II. EVALUATION MEASUREMENT FOR THE SIX ALGORITHM

Algorithm	Execution Time	Sensitivity	Accuracy
LR	0.047	99.70	89.73
DT	0.071	94.24	98.41
NB	0.088	98.95	79.98
SGD	0.042	94.41	88.66
KNN	28.255	92.15	99.05
RF	3.795	99.77	99.13

We compared these algorithms to determine the best algorithm in terms of the three parameters, such as the highest execution accuracy, higher sensitivity, and less execution time as shown in Fig. 3, Fig. 4 and Fig. 5.

Based on the analysis conducted, it was found that the random forest algorithm exhibited higher accuracy and sensitivity as compared to the other algorithms, but its execution time was relatively longer. To address this issue, an enhanced random forest classifier will be developed to maintain high levels of accuracy and sensitivity while reducing execution time. The proposed method is designed to reduce the effects of DDoS and Delay attacks. It incorporates a multi-layered approach to detect these attacks.

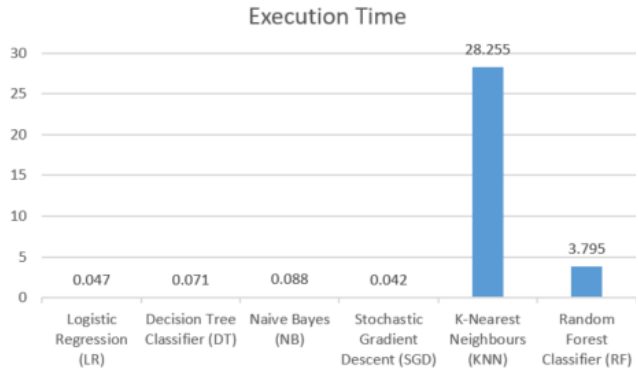

Figure 4. Execution Time of the six algorithms.

2) Second Experiment

Execution time is an important consideration when developing machine learning models for DDoS and Delay attacks detection because it directly affects the ability of the model to quickly and accurately classify incoming traffic. Since DDoS and Delay attacks often involve high volumes of traffic, models must be able to process incoming data in near-real-time to effectively detect and mitigate the attack.

Therefore, we proposed an Enhanced Random Forest Classifier for Achieving the Best Execution Time (ERF-ABE) in order to increase the overall performance of the machine-learning model that was used to predict whether a node was currently under attack or not. To be more specific, we improved the algorithm by changing a few of algorithm parameters in order to reduce the execution time and in the same time keeping the values of accuracy and sensitivity. In addition, we used Principal Component Analysis (PCA) feature for improving and

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

keeping the accuracy value because it is utilized to lessen the dimensionality of high-dimensional datasets by determining the characteristics that are most significant which helps us increase the performance of the model and reduce the amount of time it takes to execute by reducing the number of features.

The parameters that we using for tuning the algorithm are: Estimators, Max samples, Max feature, Maximum depth.

Through experimentation, we observed that modifying the parameters of the Random Forest Classifier and applying Principal Component Analysis (PCA) to our dataset, the execution time has been reduced. Fig. 6 illustrates the achieved reduction in execution time without compromising the high accuracy and sensitivity, where the execution time in the ERF-ABE has been reduced by (3.116) seconds compared to the default random forest classifier and the values of both accuracy and sensitivity remained almost the same by 99%. The results proved that careful selection and tuning of algorithmic parameters, associated with dimensionality reduction techniques such as PCA, could effectively enhance the RF's performance in detecting attacks on network nodes.

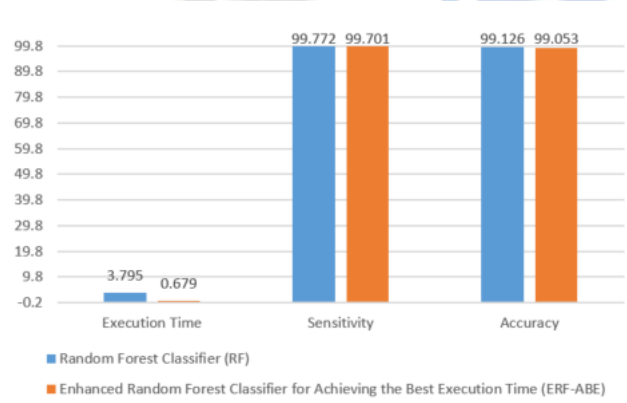

Figure 6. Comparison between Random Forest Classifier and Enhanced Random Forest Classifier.

V. CONCLUSION

The proposed ERF-ABE algorithm provides a robust and reliable approach for securing the IoMT against DDoS and Delay attacks. The ERF-ABE ensures the best accuracy and sensitivity while minimizing the execution time. It can be applied to a variety of medical applications, ensuring the safety and privacy of patient data. Further research is needed to evaluate the effectiveness of the proposed ERF-ABE under different attack scenarios.

Highly accurate machine learning models can prove exceptionally effective in detecting attacks on IoMT devices, particularly when trained on extensive datasets containing instances of both attacks and non-attacks. Following the model's training, it can be deployed in real-time to actively monitor IoT devices and identify any potential signs of suspicious activity.

Future work will be focus on developing a methodology to prevent attacks on IoMT devices by monitoring the network traffic and behavior of the IoT devices for signs of compromise or suspicious activity, and taking action to block or isolate any devices that are found to be compromised.

ACKNOWLEDGMENT

Appreciation and thanks to everyone who contributed to the completion of this work.

REFERENCES

[1] Vishwakarma R, Jain AK. A survey of DDoS attacking techniques and defence mechanisms in the IoT network. Telecommunication systems. 2020;73(1):3-25.

[2] Gaur V, Kumar R. Analysis of machine learning classifiers for early detection of DDoS attacks on IoT devices. Arabian Journal for Science and Engineering. 2022;47(2):1353-1374.

[3] Haque MR, Tan SC, Yusoff Z, Nisar K, Lee CK, Kaspin R, et al. Automated controller placement for software-defined networks to resist DDoS attacks. Computers, Materials & Continua. 2021.

[4] Ali MH, Jaber MM, Abd SK, Rehman A, Awan MJ, Damaševičius R, et al. Threat analysis and distributed denial of service (DDoS) attack recognition in the internet of things (IoT). Electronics. 2022;11(3):494.

[5] Abdullah A, Hamad R, Abdulrahman M, Moala H, Elkhediri S. CyberSecurity: a review of internet of things (IoT) security issues, challenges and techniques. In: 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS). IEEE; 2019. p. 1-6.

[6] Yaqoob I, Hashem IAT, Ahmed A, Kazmi SA, Hong CS. Internet of things forensics: Recent advances, taxonomy, requirements, and open challenges. Future Generation Computer Systems. 2019;92:265-275.

[7] Lee, B.-K. . (2023). A Study on Image Quality Improvement for 3D Pagoda Restoration. International Journal of Intelligent Systems and Applications in Engineering, 11(4s), 150–156. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2582

[8] Mohamed MB, Meddeb-Makhlouf A, Fakhfakh A, Kanoun O. Robust Jamming Attacks Detection Algorithm for Healthcare Applications. In: 2022 19th International Multi-Conference on Systems, Signals & Devices (SSD). IEEE; 2022. p. 1333-1340.

[9] Al-Hadhrami Y, Hussain F. K. DDoS attacks in IoT networks: a comprehensive systematic literature review. World Wide Web. 2021;24(3):971-1001.

[10] Dr. Avinash Pawar. (2020). Development and Verification of Material Plasma Exposure Concepts. International Journal of New Practices in Management and Engineering, 9(03), 11 - 14. https://doi.org/10.17762/ijnpme.v9i03.90

[11] Lou X, Tran C, Yau DK, Tan R, Ng H, Fu TZ, et al. Learning-based time delay attack characterization for cyber-physical systems. In: 2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm). IEEE; 2019. p. 1-6.

[12] Janiesch C, Zschech P, Heinrich K. Machine learning and deep learning. Electronic Markets. 2021;31(3):685-695.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6
DOI: https://doi.org/10.17762/ijritcc.v11i6.7558
***Article Received**: 04 April 2023 **Revised**: 28 May 2023 **Accepted**: 24 June 2023*

[13] Mohamed MB, Meddeb-Makhlouf A, Fakhfakh A, Kanoun O. Intrusion detection based on correlation of multiple health signals in wbsn. In: 2020 17th International Multi-Conference on Systems, Signals & Devices (SSD). IEEE; 2020. p. 372-377.

[14] Chiew KL, Tan CL, Wong K, Yong KS, Tiong WK. A new hybrid ensemble feature selection framework for machine learning-based phishing detection system. Information Sciences. 2019;484:153-166.

[15] Aljabri M, Alahmadi AA, Mohammad RMA, Aboulnour M, Alomari DM, Almotiri SH. Classification of firewall log data using multiclass machine learning models. Electronics. 2022;11(12):1851.

[16] Wang Y, Li Y, Wang X, Xiaohui Z. A novel traffic generator for switch testing. In: 2015 International Conference on Environmental Engineering and Remote Sensing. Atlantis Press; 2015.

[17] Emmerich P, Gallenmuller S, Raumer D, Wohlfart F, Carle G. Moongen: A scriptable high-speed packet generator. In: Proceedings of the 2015 Internet Measurement Conference. ACM; 2015. p. 275-287.

[18] Megyesi P, Szabo G, Molnár S. User behavior based traffic emulator: A framework for generating test data for dpi tools. Computer Networks. 2015;92:41-54.

[19] Kuwabara Y, Yokotani T, Mukai H. Hardware emulation of IoT devices and verification of application behavior. In: 2017 23rd Asia-Pacific Conference on Communications (APCC). IEEE; 2017. p. 1-6.

[20] Hsu WH, Li Q, Han XH, Huang CW. A hybrid IoT traffic generator for mobile network performance assessment. In: 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC). IEEE; 2017. p. 441-445.

[21] Pullmann J, Macko D. Network tester: A generation and evaluation of diagnostic communication in IP networks. In: 2018 16th International Conference on Emerging eLearning Technologies and Applications (ICETA). IEEE; 2018. p. 451-456.

[22] López, M., Popović, N., Dimitrov, D., Botha, D., & Ben-David, Y. Efficient Dimensionality Reduction Techniques for High-Dimensional Data. Kuwait Journal of Machine Learning, 1(4). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/145

[23] Kuang XH, Li J, Xu F. Network traffic generator based on distributed agent for large-scale network emulation environment. In: International Conference on Intelligent Science and Big Data Engineering. Springer; 2018. p. 68-79.

[24] Mohamed MB, Meddeb-Makhlouf A, Fakhfakh A, Kanoun O. Secure and Reliable ML-based Disease Detection for a Medical Wireless Body Sensor Networks. International Journal of Biology and Biomedical Engineering. 2022;16:196-206.