INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Intrusion Detection Using IDMAL Algorithm for IOT Devices

P Arul[1], N Shanmugapriya[2]*

1 Research Supervisor, Assistant Professor, Department of Computer Science, Government Arts College (Affiliated to Bharathidasan University, Trichy-24), Tiruchirappalli, 621 211, India
2 Assistant Professor, Department of Computer Science, Government Arts College (Affiliated to Bharathidasan University, Trichy-24), Tiruchirappalli, 620022, India

OPEN ACCESS

Received: 15-10-2022

Accepted: 26-03-2023

Published: 02-05-2023

Citation: Arul P, Shanmugapriya N (2023) Intrusion Detection Using IDMAL Algorithm for IOT Devices. Indian Journal of Science and Technology 16(17): 1268-1275. https://doi.org/10.17485/IJST/v16i17.2030

* **Corresponding author**.

gac.shanmugapriya@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objective: To develop a system based on fog computing to maintain the security of the user data and privacy in the IoT environment. The proposed work is intended to predict the assaults, automatically recognize known attacks, and select the appropriate defense mechanism to safeguard the private data in an IoT environment. **Methods:** The proposed approach is used to recognize a variety of intrusion methods, including DDOS, DOS, and multistage attacks developed by hackers with the explicit goal of seizing control of the entire IoT network and capturing the precious data. Initially the proposed work involves training the algorithm with a small amount of data, and when dynamic data begins to flow into IOT devices, the trained part will identify the potential attacks based on the earlier pattern built. **Finding:** The experimental evaluation is carried out for the proposed IDMAL along with two more existing algorithms ANN and Naïve Bayes to determine the performance of the proposed algorithm. The experimental results it is quite clear that the IDMAL after 12.3 seconds of training the attack detection accuracy is 97.1% with a false rate of 0.31%. **Novelty:** With a higher level of accuracy and precision, the suggested IDMAL algorithm makes use of cutting-edge methodologies to interpret attacks in advance utilizing a prediction mechanism from the IoT traffic data. The proposed algorithm is compared with two of the well-known existing algorithms namely ANN and Naïve Bayes and from the experimental result; it is quite obvious that the proposed algorithm is much more accurate and precise in detecting the attacks.

Keywords: Iot; Cloud computing; Fog computing; Security; Machine learning

1 Introduction

Many studies are carried out in the past related to the security and the privacy of the IoT devices and the most important limitations and snags are discussed in this introduction section. The attacks are either detected or predicted [(1)] using many notions using deep learning where the entire system works as client server model where the nodes train the model to detect the assault [(2)] and these nodes will utilize several parameters to get

accurate detection but the limitation here is the time consumption and not suitable for live feed present in the IoT environments. Deep learning has also successfully been implemented in various fields, proving its superiority in tackling intrusion detection attacks. Due to the limitation of signature-based detection for unknown attacks, the anomaly-based Intrusion Detection System (IDS) gains advantages to detect zero-day attacks [3]. IoT security threats and challenges for IoT networks by evaluating existing defense techniques. Our main focus is on network intrusion detection systems, existing implementation tools and datasets as well as free and open-source network sniffing software [4].

Studies related to naïve Bayes algorithm to detect the multiclass attacks are analyzed but when a new data is incorporated in the naïve Bayes, it struggles to learn and takes very long time to get adapted. Also the accuracy levels of the existing algorithm during the testing are low and produces lot of false alarm rate. This is the major drawback present in the Naïve Bayes. Studies related to prediction works only on specific attacks and one particular algorithm named FORE – FOrecasting using REgression analysis uses prediction method to predict the worm attacks in the IoT but the major drawback is it can only predict only the worm attack. To overcome this attack oriented prediction, various framework are designed to forecast all types of attacks using the prior footprints but extracting this footprint data related to attacks dynamically is slow and most of the attacks occurred before the prediction due to lagging speed.

The ANN is then utilized using smart grid technique and the precious information extracted from these grids is employed to create probability distribution database. This database is used to train the neural network to predict the cause of attacks accurately. But this technique requires lot of training models and energy to accomplish the prediction task dynamically.

Most of the existing algorithm's works are limited with some specific previously known attacks and they are designed and developed to mitigate those particular vulnerabilities. If a generalized mitigation procedure is created to counter the unknown attacks it lacks the speed and accuracy.

1.1 Problem Statement

The detection of the attacks at an early stage in the internet of things is a major one and this needs to be designed and deployed quite well to avert the unwanted loss and privacy to the user's data present in the IOT network. Most of the existing methods utilizes some of the existing algorithms like ANN and Naïve Bayes to classify and detect various attacks like DoS (SMURF, Tear drop, and Neptune), U2R (Overflow and rootkit) and R2L (guessing, Imap and Multihop) present in the system only after the attack occurred but it was highly impossible to detect the attacks in advance, also the accuracy rate of the detection is not up to the mark and the false rate is quite high in these existing methods [5]. To overcome this accuracy related problem, the proposed algorithm is developed with a new idea of separating the entire algorithm into two stages which is clearly mentioned in the proposed approach section. The response time of the proposed algorithm when implemented in the FOG and cloud environment are compared to gauge the performance of the proposed approach and from the experimental result, it is clear that the proposed IDMAL algorithm fared reasonably well than the other two existing algorithms.

1.2 Research Gap

The main snag or the gap that is present in the existing methods is the response time to report the details of an attack that was identified and latency present in the network which provides the report late and thereby causes a huge loss to the user present in the network. The proposed algorithm comes up with a new approach to alleviate these snags and safeguard the network and prevents the intrusion in advance using the weightage scheme and checks for the anomaly in the data packet than most of the other existing approaches. Also the proposed algorithm requires very less iteration of training to achieve high response time, accuracy and speed.

1.3 Fog Computing

The fog computing is a simplified version of cloud computing which is developed by CISCO. This technology is based on distributed network where the archival, processing of data is carried out at the edge of the network. The main focus of the fog is to reduce the latency present in the network where colossal volume of data is communicated across millions of interconnected gadgets. The fog computing also consumes low bandwidth when compared to other technologies and this reduces the overall cost considerably. The issues like mobility, scalability, heterogeneity are easily handled by the fog technology.

As shown in Figure 1, the IoT application architecture based on fog computing consists of three parts, namely Gadgets, Fog, cloud. The gadget contains a basic sensor that can be connected to the Internet to transfer data from the gadget and to receive the data from the internet into the gadget. This part's primary job is to gather and collect environmental data and transmit it to the fog computing. The fog is built on distributed computing, network edge devices including access points, gateways, and routers, storage devices. Data from the fog nodes are received by the cloud, which manages all data across the globe which is

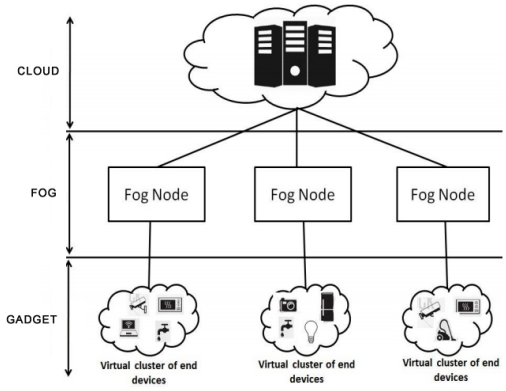

Fig 1. Architecture of fog computing

primarily decentralized [5].

In order to safeguard the IOT environment, this paper aims to design and create a system based on fog computing to preserve the security of the data. The major goal of the proposed work is to automatically detect known attacks, forecast attacks, and choose the best defense to protect the data in an IoT environment. The proposed IDMAL algorithm uses advanced techniques to interpret the attacks in advance using prediction mechanism from the IoT traffic with a higher level of accuracy and precision.

1.4 Machine Learning

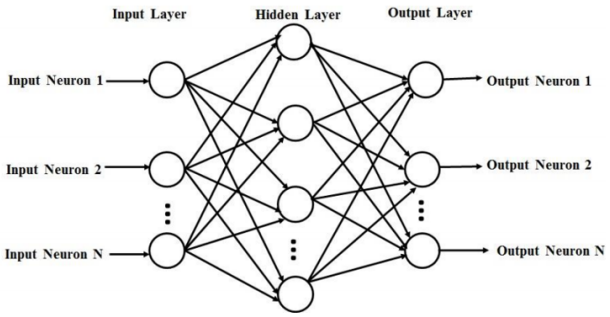

Fig 2. Traditional machine learning method

The traditional machine learning algorithms are slow and does not possess the capacity to cater to the need of processing the data dynamically and feed it back to the IoT devices and this important limitation is overshadowed in this paper using random weight choosing method using simple matrix computations.

Let us assume that for a given training set of N samples, with n attributes and m classes where the input is

$$\text{Input}_i = \left(\text{Input}_{i1}, \text{Input}_{i2,}, \ldots \text{Inputin}^T \right)^T \in R^n$$

$$\text{Output}_i = \left(\text{Output}_{il}, \text{Output}_{2,}, \ldots \text{Outputim}^T \right)^T \in R^m$$

2 Methodology

2.1 Proposed Approach

The proposed IDMAL algorithm comprises of two stages, namely

1. Initialization stage
2. Learning stage

The initialization part is training part where the algorithm got trained using optimum or minimum data. The initial hidden matrix is computed using the following formula,

$\boldsymbol{HI}_0 = [h_1,\ldots,h_n]^T$

The initial weightage matrix for the hidden layer is computed then,

$\beta_{\mathbf{0}} = \frac{1}{HI_0 T_0} HI_0{}^T T_0$

Similarly for the next training data,

$\boldsymbol{HI}_{k+1} = [h_1,\ldots,h_N]^T$

The output weightages are computed using the recursive method as,

$O_{\mathbf{k+1}} = O_k - \frac{O_k h_{(k+1)} h_{k+1}^T O_k}{1+h_{k+1}^T O_k h_{k+1}}$

Where $O_k = \frac{1}{\left(HI^T{}_k H_k\right)}$

From these computations and formula, the weightage for the hidden layer is found and this value is used to predict the attacks as the learning is made from a very small data when compared with the orthodox machine learning algorithms. This weight computed from the data is used to predict the next attack by the intruders and some sort of pervasive remedy can be taken prior to the damage created to the user data and the overall wireless network. Since the fog computing works on the edge of the cloud, the data needed to train the data is also very small. The pseudo code of the proposed algorithm is shown in the following Figure 3 where the inputs is a small data, the weight are computed to predict the nature of the attack and thereby reduces and evades the assault before it actually takes place.

Algorithm IDMAL

INPUT:

Hidden Neurons → *Ne*
Activation method →*G(x)*
Classes present → *C*

Initialization:

- Load the initial training set
- Assign initial random weights
- Compute the hidden matrix
- Compute the initial output weight

Identifying:

1. ∀ sequential data SD ∈ online *do begin*
2. Fetch the sequential data
3. Compute the latest hidden layer matrix
4. Compute the latest output weight using recursive

CLOSE For

Close

Fig 3. Pseudo code of the proposed algorithm

From the above Figure 4 the pseudo code of the proposed algorithm is given and it identifies several intrusion attacks like DDOS, DOS, and multistage attacks like probe, U2R created by the intruders purposefully to take control of the entire IoT network. The initialization part is the training part where the algorithm is getting trained with the minimum data set and then when the dynamic data starts to enter the IOT devices the learning or Identifying part will pinpoint the probable attacks from the pattern created earlier from the trained data set. This method is very simple and easy to implement in the fog and IoT environment as it needs only few data and steps to train the system to identify the intrusions. The overall working of the proposed algorithm is shown in the following Figure 5.

2.2 Proposed Methodology

The proposed methodology comprises of two stages, they are

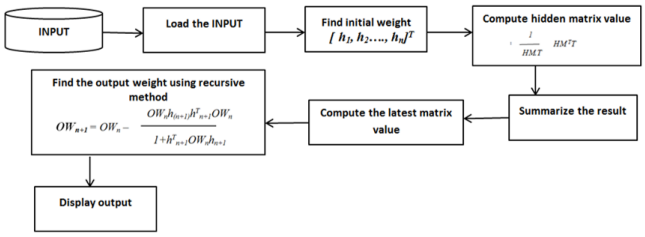

Fig 4. Flow diagram of the proposed algorithm

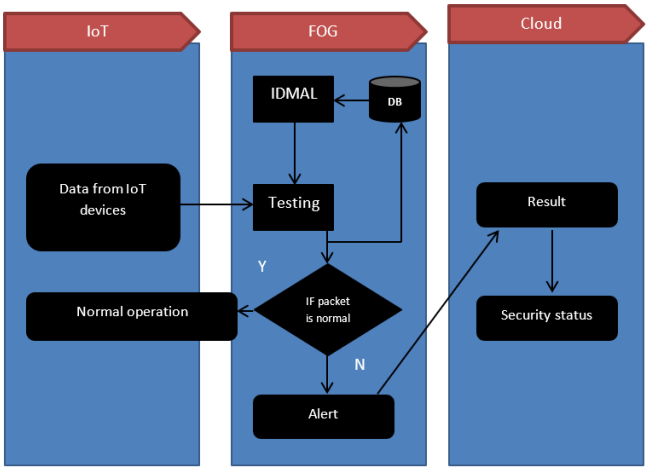

Fig 5. Overall working of the IDMAL algorithm

2.2.1 ***Initialization***

This stage actually loads the dataset into the algorithm and the data is initially scanned. The scanned data is processed to locate the hidden neurons, once the neurons are identified; they are assigned with some random weightages without leaving it empty. The matrix value is computed using the formula,

$$\beta_{\mathbf{0}} = \frac{1}{HI_0T_0} HI_0{}^T T_0$$

Once the initial matrix value is found, the final output weightages are computed in the initialization stage. This output matrix value found is used to identify the probable attacks which are previously known using the data packet comparison with that of the DB. But the proposed methodology employs the second stage to detect the unknown attacks by refreshing the output matrix dynamically.

2.2.2 ***Detection***

The second stage computes the latest and current weightage for each and every data. From this weightage data, the current output matrix value is computed dynamically to address the most important problem – detection of unknown attacks accurately without large training. The current matrix value is found repeatedly using recursive method and once an anomaly is identified in a particular node an alert or warning is popped up and the attack type along with the intrinsic details related to the stack are stored in the DB.

2.3 Experimental Evaluation

The system used here is Intel I7 processor with 4GB RAM as fog nodes and the cloud used here is AZURE cloud service. The algorithm is implemented using MATLAB. The activation function is initially identified using different kernels like sigmoid and RBF. The dataset used in the evaluation is NSL-KDD which is a benchmarked dataset specifically used to analyze the intrusion attacks. The evaluation is carried out with different size of hidden layers and chunk size and from the experimental results the

sigmoid function is found to be efficient related to the accuracy in detecting the attacks and the consumes less training time when compared with the RBF.

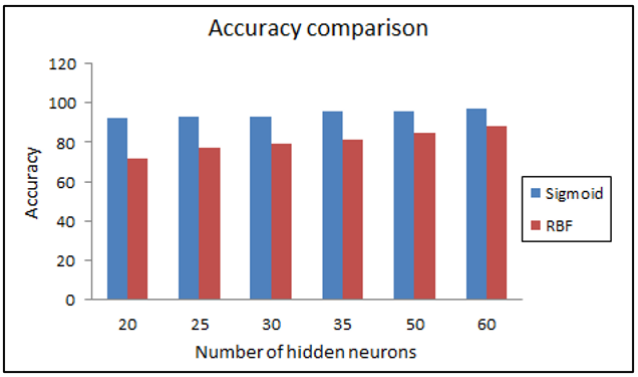

Fig 6. Accuracy comparison for various kernels

Table 1. Accuracy comparison with varying hidden neurons and chunks

Accuracy comparison with varying neurons and chunk				
Number of neurons	**Chunk size**			
	500	**1000**	**2000**	**3000**
5	56.9	58.9	57.5	57.2
10	58.7	59.2	58.8	60.2
15	67.3	67.2	66.0	68.2
20	74.7	75.1	73.9	74.6
30	82.6	80.1	83.2	83.7
40	88.3	86.3	85.6	86.2
50	92.6	90.3	90.2	91.6

Table 2. Training time comparison with varying hidden neurons and chunks

Training time (sec) comparison with varying neurons and chunk				
Number of neurons	**Chunk size**			
	500	**1000**	**2000**	**3000**
5	5.3	8.2	46.4	163
10	5.6	8.5	53.7	189.5
15	5.9	9.2	67.9	227.5
20	6.3	10.8	89.3	278.8
30	6.9	12.1	93.8	290.8
40	7.2	13.6	110.5	303.7
50	7.5	15.3	125.8	336.7

From the Tables 1 and 2, it was found that the best possible size of the hidden neurons and the chunk size is 30 and 1000 respectively and the training algorithm utilizes the neuron size of 30 and chunk size of 1000 to get the best possible results in terms of accuracy.

The overall accuracy is computed from the confusion matrix which is calculated for two algorithms along with the IDMAL are artificial neural network, and naïve Bayes algorithms. The proposed algorithm achieves an accuracy of 97.1 % with a false rate of 0.31% in appropriately 12.3 seconds of training.

The single classification result is shown in the following Table 3.

Table 3. Classification accuracy of the algorithms

Algorithm	Accuracy	Detection rate	False rate
ANN	94.2%	95.8	4.67
NAÏVE BAYES	89.35%	91.8	9.78
IDMAL	97.3%	97.8	0.32

The proposed IDMAL achieves an accuracy of over 97 percent and the false detection rate is less than 0.5% which is very superior when compared with the other two algorithms.

The response time is computed for the three works on both FOG environment and a detailed comparison was carried out. The proposed algorithm performed reasonably well for various bandwidths like 50, 250, 500, and 1000 Mb/s. The comparison graph was shown in the following Figure 7.

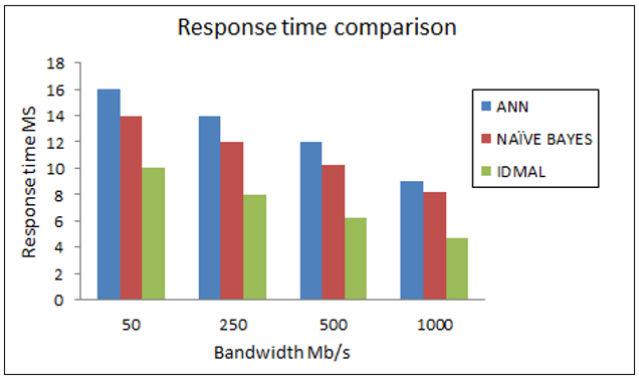

Fig 7. Comparison of response time

From the above Figure 7, the response time to report an attack is almost 20 – 30% less in the proposed IDMAL algorithm when measured with varying bandwidths and also the latency is measured to gauge the overall performance of the proposed algorithm.

3 Results and Discussion

Since the proposed IDMAL algorithm employs the weightage on the data it is fed into, it is quite easy to segregate the normal and anomaly transaction that are being carried out in the system. The benchmarked dataset used here in the evaluation is NSL-KDD dataset which comprises of 20 multi-attacks attacks as shown in the Table 4.

Table 4. Multiclass NSL-KDD dataset

Data	Trained	Tested
Without attack	63878	9710
With DOS	45109	7328
With Probe	10989	2871
With R2L	925	1692

To find the detection rate and the speed, the proposed IDMAL and other two algorithms are executed in Azure cloud service and from the experimental results, and the comparative true positives and false positives are noted for the benchmarked dataset as shown in the Table 5.

The TPR – true positive rate & the FPR false positive rate are computed for the existing and proposed algorithm and from the comparative table, it is obvious that the proposed performed well and it requires less training when compared with the other algorithms as well as it can deal with new intrusion data for learning due to its two stage processing incorporated in the working.

Table 5. Comparative true positives and false positives benchmarked dataset

Algorithm	Usual		DOS		Probe		R2L	
	TP	FP	TP	FP	TP	FP	TP	FP
Naïve Bayes	80.21	7.19	89.21	4.66	77.9	5.67	73.29	7.22
ANN	90.21	1.23	92.2	2.78	80.12	4.20	74.20	2.51
IDMAL	95.28	0.27	95.6	1.37	86.9	1.09	76.20	0.42

This enhances the overall performance of the proposed algorithm with respect to detection rate and false rate. From the experimental result, it is quite obvious that the proposed algorithm achieved better attack detection and anomaly detection when compared with the other two existing algorithms as the proposed achieved 57 to 92 percent accuracy when the chunk size of the data is increased from 500 to 3000 along with the increase in the number of hidden neurons from 5 to 50. But the naïve Bayes showed peak accuracy level of 88.6 and ANN reached its highest accuracy of 87.3 which is almost 10 percent less than the proposed IDMAL algorithm. The training time taken by the proposed along with the existing ANN and Naïve Bayes is computed and the proposed IDMAL took 5 seconds to 340 seconds depending upon the varying number of neurons and varying chunk size but the ANN took 6.7 seconds to 430 seconds to get it trained. The Naïve Bayes performed the worst as it took 7.8 seconds to 572 seconds to get it trained to get the desired output. Also from the Tables 1 and 2, it is found that the varying number of hidden neurons are much important than the varying number of chunks in the experimental dataset.

4 Conclusion

According to the experimental findings, the fog nodes identify attacks 30% more quickly than cloud-based implementations and with a lower proportion of false detection and false positives. An initial step in creating an intrusion detection system for an Internet of Things application employing fog computing is provided in this work with an algorithm which utilizes less training data and less training time. The false rate in the attack detection is 40 – 60% less in the proposed algorithm when compared with the existing algorithms. The accuracy and the detection rate is almost 10 – 20% higher than the existing algorithms. The only limitation of the proposed algorithm is it is designed to identify only the known attacks and it should be trained at least with a small data to identify the new attacks. The weightages computed in the proposed system is purely based on the known attacks and in future the system can be developed and designed to allocate weightages automatically based on the anomaly present.

References

1) Alaba FA, Othman M, Hashem IAT, Alotaibi F. Internet of Things security: A survey. *Journal of Network and Computer Applications*. 2017;88:10–28. Available from: https://doi.org/10.1016/j.jnca.2017.04.002.
2) Ammar M, Russello G, Crispo B. Internet of Things: A survey on the security of IoT frameworks. *Journal of Information Security and Applications*. 2018;38:8–27. Available from: https://iranarze.ir/wp-content/uploads/2018/02/E5779-IranArze.pdf.
3) Alsoufi MA, Razak S, Siraj MM, Nafea I, Ghaleb FA, Saeed F, et al. Anomaly-Based Intrusion Detection Systems in IoT Using Deep Learning: A Systematic Literature Review. *Applied Sciences*;11(18):8383. Available from: https://doi.org/10.3390/app11188383.
4) Chaabouni N, Mosbah M, Zemmari A, Sauvignac C, Faruki P. Network Intrusion Detection for IoT Security Based on Learning Techniques. *IEEE Communications Surveys & Tutorials*. 2019;21(3):2671–2701. Available from: https://doi.org/10.1109/COMST.2019.2896380.
5) Kumar P, Gupta GP, Tripathi R. A distributed ensemble design based intrusion detection system using fog computing to protect the internet of things networks. *Journal of Ambient Intelligence and Humanized Computing*. 2021;12(10):9555–9572. Available from: https://doi.org/10.1007/s12652-020-02696-3.