Volume 1, Issue VII, June 2022, **No. 9, pp. 101-114**
Submitted 2/6/2022
Final peer reviewed 31/6/2022
Online Publication 12/7/2022
Available Online at http://www.ijortacs.com

DEVELOPMENT OF MULTI LEVEL INTRUSION DETECTION SYSTEM FOR CLOUD BASED LOG MANAGEMENT USING MACHINE LEARNING TECHNIQUE

[1]Oduah O., [2]Olofin B.B., [3]Asogwa T.C.
[1,2,3]Enugu State University of Science and Technology
Oduahobinna621@gmail.com

Abstract

This paper presents development of a Multi Level Intrusion Detection System (MLIDS) for cloud-based log management using machine learning technique. The research methods are data collection, data processing, multi layered neural network, threat detection algorithm, threat control algorithm and multi level intrusion detection system developed with the threat algorithms. The algorithms were modeled using structural methods which engaged Universal Modeling Language (UML) diagram for the system design. Pseudocodes were also used to present the generated algorithmic outputs which were used to model the MLIDS. The system was implemented with Simulink, tested and validated with Mean Square Error (MSE) and Regression (R). The result of MSE is 2.47e-05Mu and R= 0.99445. The implication of the results showed that the new algorithm developed was able to correctly monitor, detect and prevent threat penetration to cloud-based server. A comparative analysis was also conducted with other threat detection algorithms and from the result, it was observed that the performance of the new system was better due to its multi layered configuration of neurons to enhance data processing and computation in the hidden layers, then right choice of activation used, training algorithm adopted and also the quality of data used to train the neural network and achieve the threat detection algorithm. These features of the new algorithm make it to standout from the others with better performance.

Keywords: MLIDS, Cloud, UML, Neural Network, MSE, Regression, Algorithm

1. INTRODUCTION

Over the years, various cloud-based companies have emerged with the sole responsibility of managing multi-tier enterprises data and resources all over the world. However, most services which are based on cloud infrastructure are vulnerable to attacks. This is usually because the firewalls developed for security are only focused on securing the packets and logging to the server and not the server itself, therefore, leaving it vulnerable to threat. This has remained a major problem waiting to be solved. The benefit of solving the problem will improve the integrity, data confidentiality, accessibility and reliability of cloud data management enterprises.

Many works have been developed over the years to solve the problem of intrusion on cloud based collaborative platforms, however despite their success on the security of packet penetration to the cloud, the infrastructures remain vulnerable to attack and has remained a major gap waiting to be addressed over the years. This problem will be solved in this research, developing a multi-level intrusion detection system which employed machine learning technique to secure the cloud log management server and guarantee data confidentiality and integrity. The systematic review in table 1 presented some of the existing techniques used for the protection of cloud infrastructures and their limitations or contribution to knowledge.

Table 1: Systematic review of relevant literatures

Author	Title	work done	Research gap/limitation
Uday and Manal (2016)	Fully automated deep packet inspection and verification system with machine learning.	The study used machine learning algorithm such as the K-Nearest Neighbour (K-NN), Support Vector Machine (SVM), logistic regression and Naive bayes, K-mean clustering.	The result showed that machine learning solution to cyber security are promising with high accuracy, however the study never considers flood attack
Phillip et al. (2018)	Supervised machine learning bot detection system to identify social twitter bots	The study engaged multiple ML algorithms like K-Nearest Neighbour (K-NN), Support Vector Machine (SVM), logistic regression and Naive bayes, K-mean clustering for the detection of botnets	The solution when tested showed high detection accuracy, but flood attack was not considered in the study.
Shailendra et al. (2017)	Threat detection based on ML approaches	The solution considered K-Nearest Neighbour (K-NN), Support Vector Machine (SVM), logistic regression and Naive bayes, K-mean clustering.	The result showed good threat detection performance of comparative threat datasets;however, the performance can be improved with artificial neural network.
Solomon et al. (2017)	ML predictive analysis to SQL injection detection and prevention of web-based application	The study developed a ML based security solution to protect cloud-based platforms.	The result when tested showed good result, but flood attack was not considered.

	systems.		
Doyen et al. (2019)	Malicious URL Detection System Using ML	The study considered the various ML algorithms which have been adopted to solve the problem of intrusion detection system such as Artificial neural network, K-Nearest Neighbour (K-NN), Support Vector Machine (SVM), logistic regression and Naive bayes, K-mean clustering	The study also revealed that neural network achieved the best result when compared to others.

2. METHODOLOGY

The methodology used for the development of the new system is the Top-down Design Methodology (TDDM). The TDDM was adopted to guide the system development starting from the problem down to the solution. This formulated the mode of the problem and then developed a machine learning based solution to the problem using artificial neural network to develop a threat detection algorithm and implemented on a Simulink platform. The research methods are data collection, data processing, artificial neural network, training and multi-level intrusion detection system.

2.1 The Case Study Cloud Platform

The case study cloud log management system considered for this research is the Alfresceo catering service limited and located at Garki, Abuja, Nigeria. The company is responsible for the management of multiple enterprise information over there cloud log management platform with services such as document management, enterprise collaboration, analytics and insights on process management, open-source enterprise solutions for content management capabilities, etc. This company was selected as a case study for this research due to the huge volume of information they management and hence made them a primary target for criminals and attackers. The geographical coordinates of the company are Lat 6.55717N; Long 3.36400E as shown in the figure 1;

Figure 1: Location of the Case study (Courtesy: Google Map)

2.2 Data Collection

The main source of data collection used for this research is the Afresco Service Limited. The company provided 12gigabyte sample of their log management server records from 4th to 27th April, 2022. This data was used as the training dataset for the machine learning algorithm to be developed shortly. The description of the data collected was presented in the table 2.

Table 2: Data Attributes of Log Files

Attributes	Description
Client IP	The IP address of the client machine
Client name	The name of the client
Data	Data of user access
Time	Time of user log
Server site	The name of the internet service as appeared on the server
Server computer name	Server name
Server IP addresses	The IP of the internet service provider
Serer port	Server port used for the data transmission
Client server URL stream	Targeted default web page of the site
Client server URL query	The client query which begins with "?"
Server client status	the status code returned by the user link
Server client size	The size of data transmitted in bytes
Client server method	Client methods of request like Get, POST or HEAD
Latency	Total time taken for client to perform action
Client server version	Protocol version like the HTTP
Client server host	Host header name
User agent	Browser type used by the client
Cookies	Contents with cookies
Referrers	Link from where clients jump to the site
Server client win32 status	The window status code

2.3 Data Processing

Due to the huge volume of the data managed by the server and collected for the research, the need for processing was vital to remove bugs, noise and other unnecessary features which might compromise the new system security integrity. The data processing was done using associate rule-based mining algorithm adapted from (Mohd et al., 2008) to sorts and removes data formats with jpg, gif, bmp, etc to ensure quality data for training.

2.4 Machine Learning Algorithm

The nature of the problem been solved is a pattern recognition type and hence Machine Learning (ML) algorithm is the appropriate solution to it. The ML is set of mathematical algorithms which can learn data and then solve pattern recognition problems. The algorithm is trained to learn and then develop the desired model for the detection

of the user log information. The machine learning algorithm adopted to solve this problem is the artificial neural network.

2.5 Artificial Neural Network (ANN)

ANN is a machine learning algorithm biologically inspired from the model of the human brain. These ANN are built with neurons which have weights, bias function and activation functions to learn data and then solve problems. In this research, the neural network model in Ashigwuike et al. (2020) was adopted and used to develop a security algorithm which identified threats from the incoming data request to the log server intelligently.

2.6 Threat Detection Algorithm

The previous section discussed the threat detection algorithm which was developed by the neural network. Now that the threat was detected, the control algorithm was used to ensure that the threat do not penetrate to the sever and cause problem. This was achieved using simple rule-based logic control algorithm which makes decision from the output of the threat detection classification to allow throughput or deny it.

2.7 Multi-Level Intrusion Detection System

The security of the log management server involved two steps which are the detection to detect the problem and then the control to isolate it from the targeted severs. This was achieved using the ANN and rule base logic control algorithm to train all incoming log request to detect threat and prevent it from the targeted server. The detection algorithm monitors all data logs to the server and when abnormally is detected, the control enables access denial to the log and hence protect the server.

3. SYSTEM DESIGN

This section presents the problem formulation, modeling of the multi-level intrusion detection system and the complete system modeling.

3.1 Modeling of the Problem Formulation

The problem formulation considered the IP flood attack which is a form of denial-of-service attack used by hackers to target cloud-based infrastructures and initiate ransomware. The architectural model in figure 2 presented the workflow of the attack penetration.

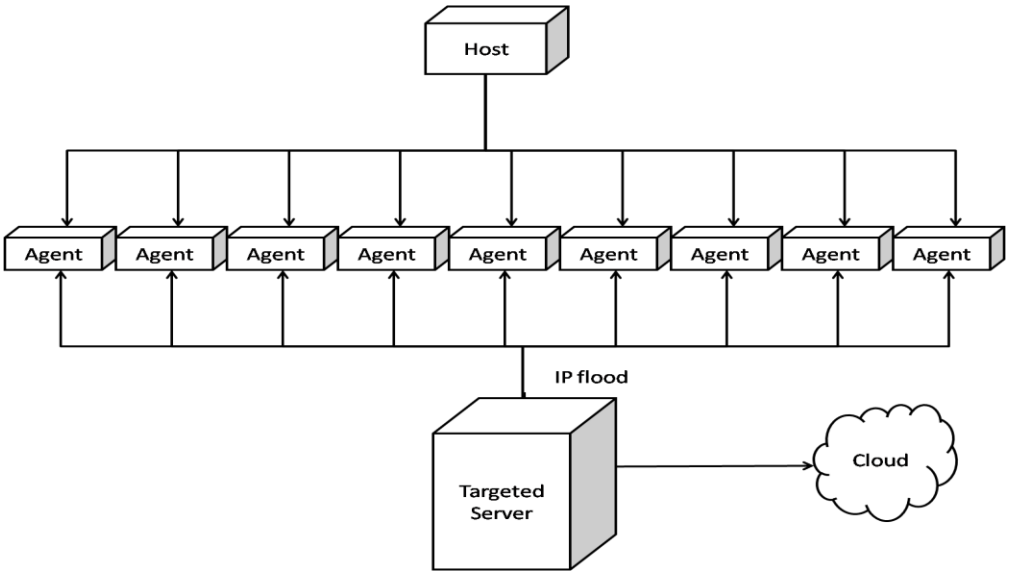

Figure 2: Architectural model of the problem formulation

In the figure 2, the attacker (host) created multiple agents which help in the distribution of the threat simultaneous to the targeted host. When this happens the host server is overwhelmed and then suffers congestion problem and eventual shutdown.

3.2 Modeling of the Neural Network Algorithm

The neural network algorithm was formulated from a single neuron as shown in the architectural model in figure 3;

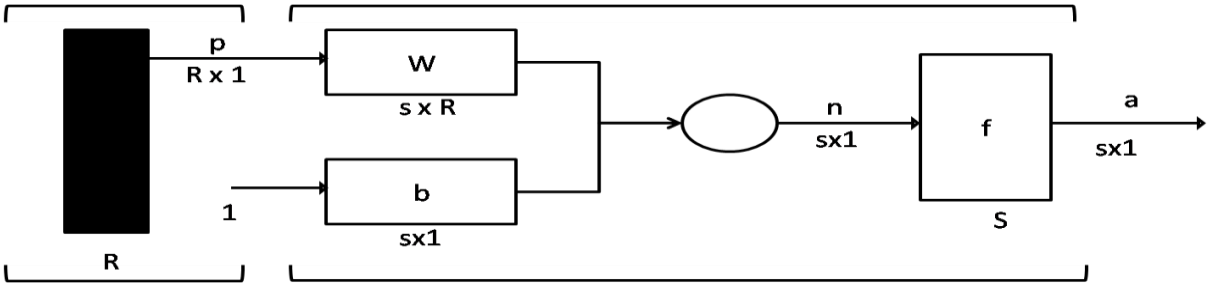

Figure 3: model of a single neuron

Where a is the output of the neuron after activation, p is the input vector, W is the neuron weight, b is the bias function of the neuron, n is scalar output, R is the number of elements in the input vector, s is the number of neurons in the input layer. The neuron in figure 3 was interconnected to form the multi layered neural network architecture in figure 4;

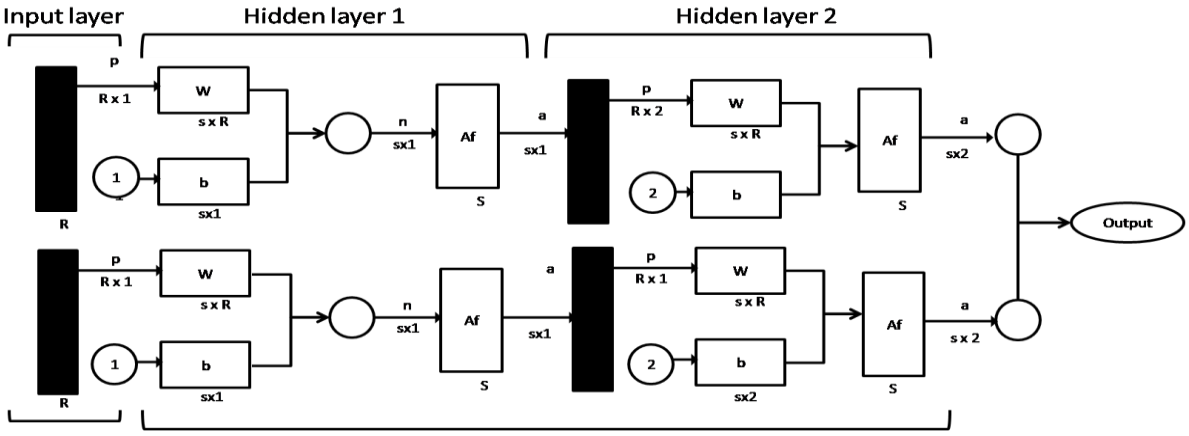

Figure 4: The architectural model of the neural network

The figure 4 presented the architectural model of the neural network algorithm which was configured to train the data collected from the log server. The neural network algorithm was developed with specified number of neurons based on the data attributes of the log server input, Rectified Linear Unit (ReLU) as the activation function (Af) and back propagation algorithm. The ANN algorithm was loaded with the data and then trained to generate the detection algorithm. The neural network training model was presented in figure 5;

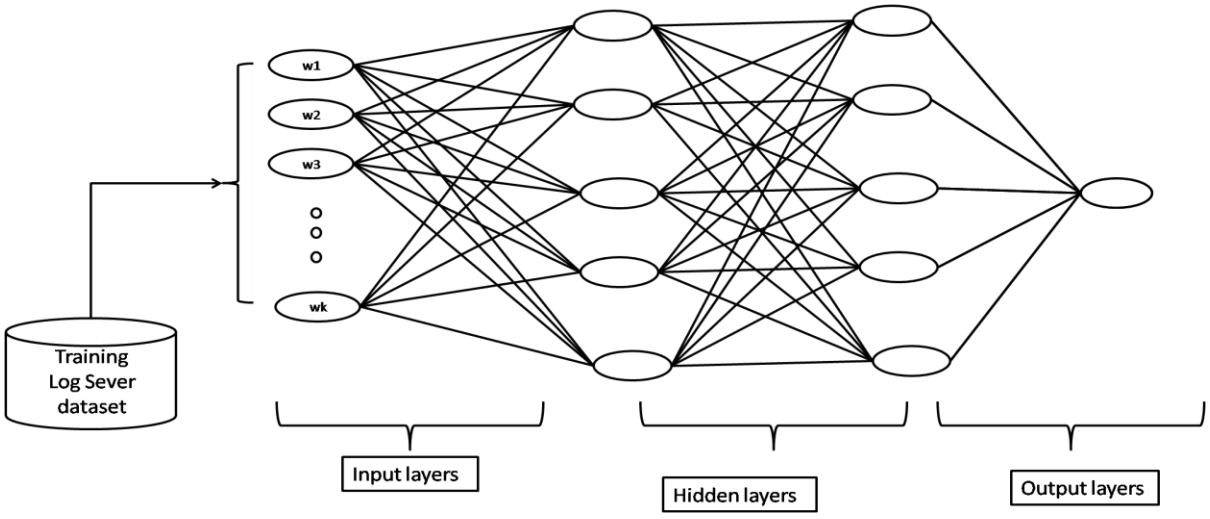

Figure 5: The neural network training model

The figure 5 presented the neural network model which was used to train the data collected to generate the detection algorithm. Before the training, the neural network splits the training data into training, test and validation set. The training was used to learn the neurons of the log patterns on the server, the testing was used to test the regression performance and then the validation was used to ensure reliability in result. During the training process the neurons are adjusted to learn the log in patterns on the server as various time logs, iteratively using a back-propagation training algorithm until a constant output was achieved. The pseudocode of the threat detection algorithm was presented as;

3.3 The threat detection pseudocode (Algorithm 1)

1. Start
2. Load log server data
3. Divide data into training and target set
4. Configure neural network
5. Initiate training algorithm
6. Initiate activation function
7. Set training epoch values and interval
8. Train neural network
9. If
10. Training is completed = true
11. Generate threat detection algorithm
12. Else
13. Return to training algorithm
14. End if
15. End

3.4 The model of the threat control algorithm

The model developed in the figure 5 presented the threat detection architectural diagram. This was used for the monitoring and detection of threat log to the server. However, to control the problem so as to prevent it from passing through the cloud server and not to infect the infrastructures, the flow chart in figure 6 was used to model it as;

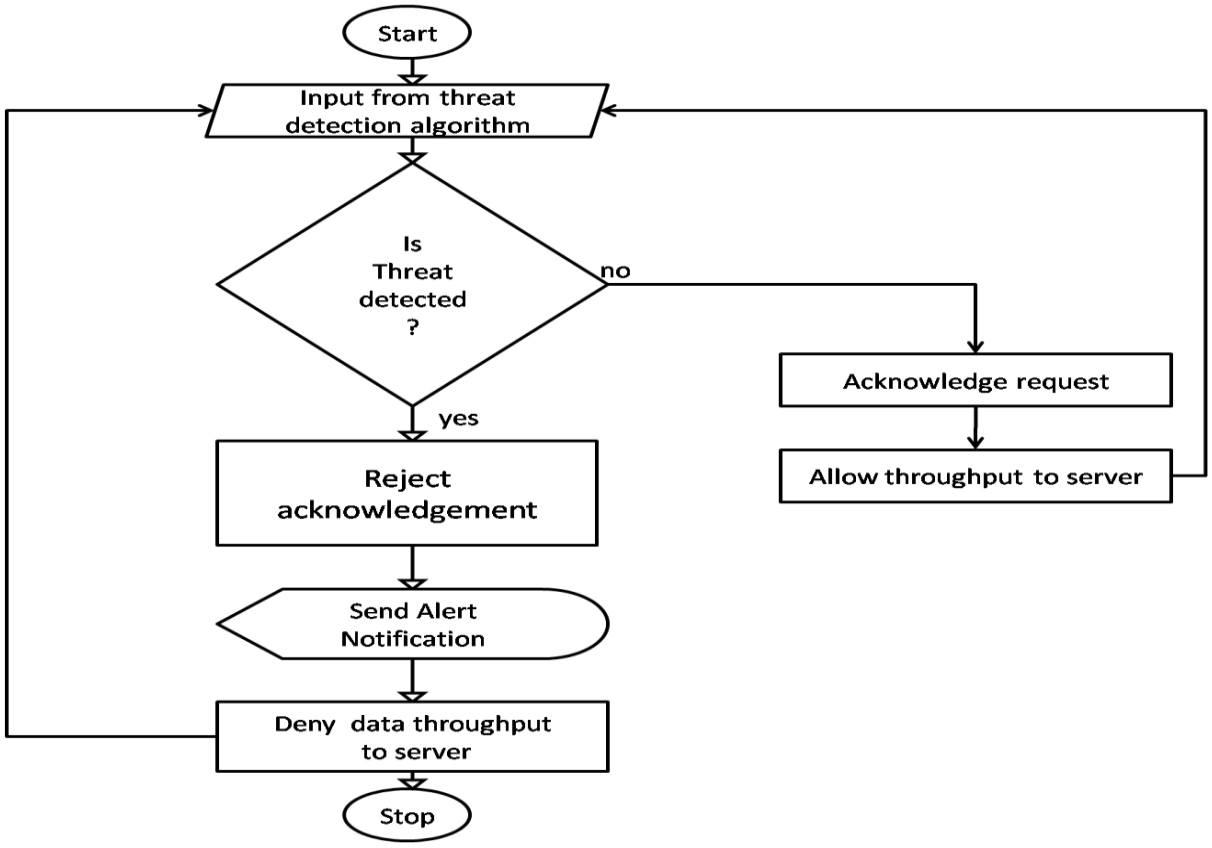

Figure 6: The flow chart of the threat control algorithm

The figure 6 presented the threat control algorithm which identified the output from the monitoring and detection algorithm and

then if threat was detected isolate it from the server to prevent shutdown. The pseudopodium of the control algorithm is presented as;

3.5 The model of the threat control pseudocode (Algorithm 2)

1. Start
2. Identify input from algorithm 1
3. If
4. Threat is detected = true
5. Log Server Rejects acknowledgement of packet
6. Send alert notification
7. Deny data throughput to cloud
8. Else
9. Acknowledge request
10. Allow throughput to cloud
11. End if
12. Return to algorithm 1
13. End

3.6 Development of the multi level intrusion detection system

The model of the multi level intrusion detection system was developed to monitor, detect and control the penetration of threat on the log manager. This was achieved using the threat detection algorithm and the control algorithm to develop a multi level intrusion detection system. The model of the new system is presented in the flow chart of figure 7;

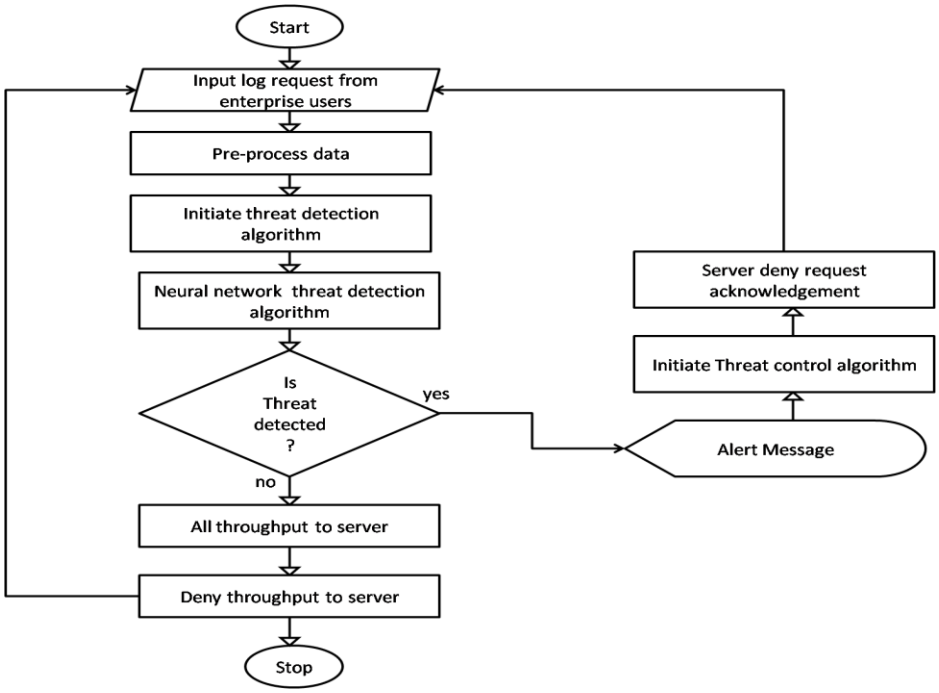

Figure 7: flow chart of the multi level intrusion detection system

The flow chart in figure 7 presented the workflow of the intrusion detection system, showing the logical relationships between the various modules which interacted to achieve the complete system. The data input from the user enterprises as logged into the server, the adopted processing algorithm removed noise with image formats which is common with such data and then the algorithm 1 was used to train and detect threat for control via access denial to the target using the threat control algorithm, else when threat was not detected, throughput is allowed to the server.

4. IMPLEMENTATION

The system designed was implemented with system identification toolbox, statistics and machine learning toolbox, optimization toolbox, neural network toolbox, communication toolbox and Simulink. The system identification toolbox was used by the neural network to load the training dataset. The system designed was implemented with system identification toolbox, statistics and machine learning toolbox, optimization toolbox, neural network toolbox, communication toolbox and Simulink. The system identification toolbox was used by the neural network to load the training dataset.

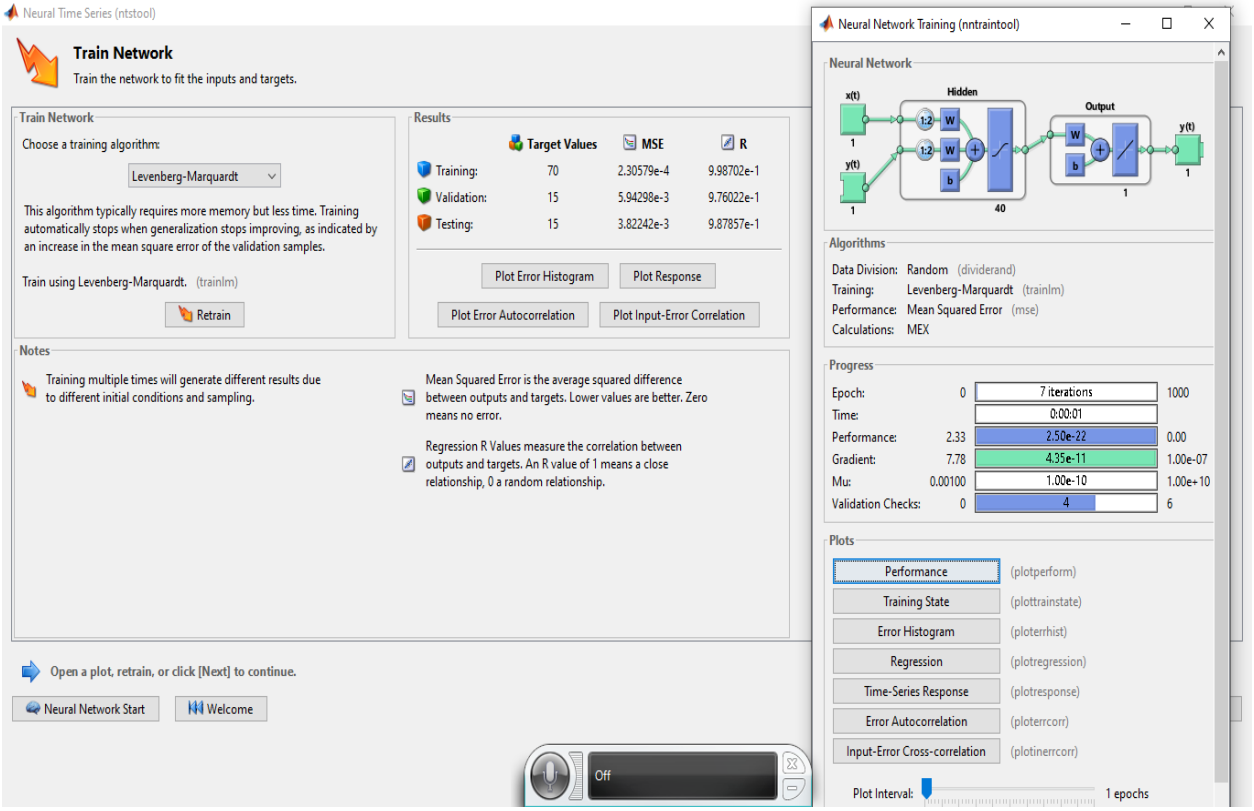

Figure 6: The neural network training tool

The figure 6 shows the neural network training tool used to train the neurons of the data collected to generate the threat detection algorithm. This algorithm was married with the threat control algorithm using basic MATLAB script programming.

5. SYSTEM TESTING

This section presented the performance evaluation of the intrusion detection system developed. To evaluate the performance, means square error, regression and validation models in Amirreza (2012) were adopted. The regression result was presented in figure 7;

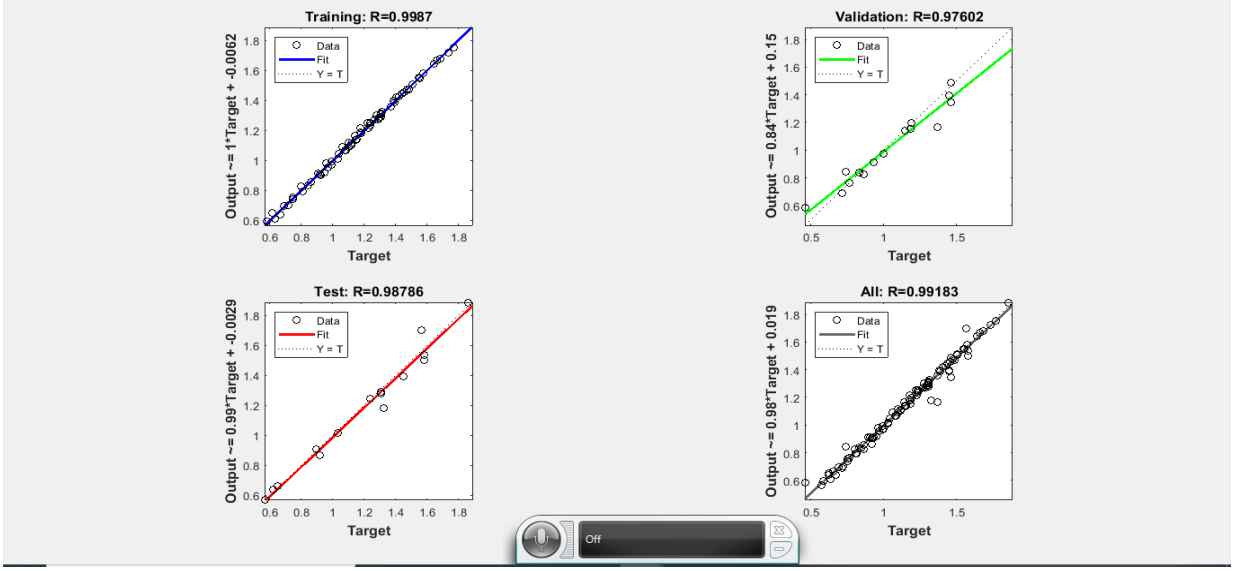

Figure 7: Regression performance of the neural network algorithm

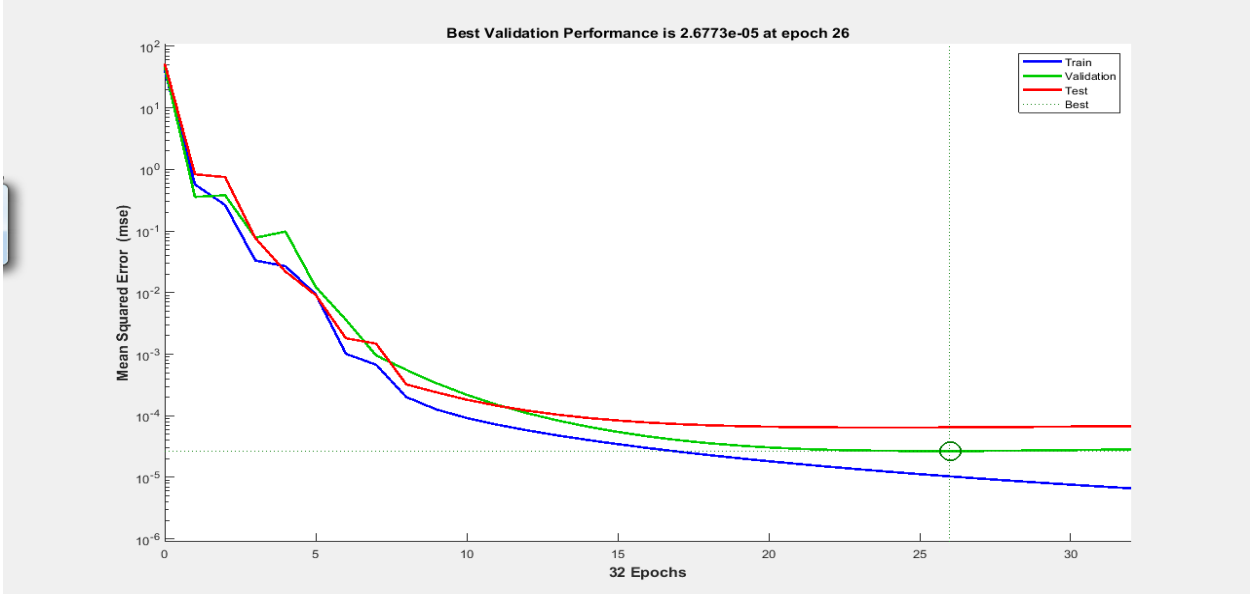

Figure 8: MSE performance

The MSE result in figure 8 presented the error achieved in the neural network training process toward the development of the security model. The aim of the researcher here is to achieve MSE of approximately zero which implied tolerable error during the training process.

From the result achieved, the MSE is 2.6773e-05Mu. The implication of this result showed that the training process was very good as the error achieved is approximately zero. To further justify these results achieved, the tenfold cross validation was used which iteratively evaluated the MSE

and R in ten-folds and the average reported as the overall system result.

5.1 Comparative Analysis

The comparative analysis drew insight from a recent work on qualitative analysis of machine learning based intrusion detection system by Binita (2021) and compared the new multi-intrusion detection system developed with the selected algorithm in the study as shown in table 4

Table 4: Comparative Analysis

Regression algorithms	Regression
Logistic regression	0.730
Decision tree	0.980
K-NN	0.950
Random forest	0.9800
GausianNB	0.9600
Multi-level intrusion system	0.9945

The table 4 presented a comparative analysis of the new system developed with the existing systems. From the result, it was observed that the performance of the new system was better due to its multi layered configuration of neurons to enhance data processing and computation in the hidden layers, then right choice of activation used, training algorithm adopted and also the quality of data used to train the neural network and achieve the threat detection algorithm. These features of the new algorithm make it to standout from the others with better performance.

6. CONCLUSION

This research has successfully developed a multi level intrusion detection system for cloud-based log management system using machine learning technique. The researcher developed a neural network-based security algorithm for the detection of intrusion. Rule based approach was used to control the detected threat so as to isolate it from the cloud. The two security algorithms developed were married as multi level intrusion detection system and implemented with Simulink. The result after testing and validation recorded MSE of 2.47e-05Mu and R= 0.99445. The implication of the results showed that the new algorithm developed was able to correctly monitor, detect and prevent threat penetration to cloud-based server. A comparative analysis was also conducted with other threat detection algorithms and from the result, it was observed that the performance of the new system was better due to its multi layered configuration of neurons to enhance data processing and computation in the hidden layers, then right choice of activation used, training algorithm adopted and also the quality of data used to train the neural network and achieve the threat detection algorithm. These features of the new algorithm make it to standout from the others with better performance.

ACKNOWLEDGEMENT

The corresponding authors of this research acknowledge Dr. and Dr Mrs. F.N. Oduah for their immense support, throughout the period of this research. Engr. Dr. Emeka Okorie is also acknowledged for his guidance and counseling to make this research a reality. Finally, thanks to Destinet Smart Technologies LTD. which provided technical and intellectual support to accomplish this research.

REFERENCES

Amirreza Zarrabi, (2012). Research on Internet Intrusion Detection System Service in a Cloud, appear in International Journal of Computer Science Issues, Vol. 9, Issue 5, No 2, September 2012, ISSN (Online): 1694-0814

Ashigwuike E.C, A. R. A. Aluya, J. E. C. Emechebe and S. A. Benson (2020) " Medium Term Electrical Load Forecast Of Abuja Municipal Area Council Using Artificial Neural Network Method " Nigerian Journal of Technology (NIJOTECH) Vol. 39, No. 3, July 2020, pp. 860 – 870; Print ISSN: 0331-8443, Electronic ISSN: 2467-8821

Binita S. (2021) "Comparative Analysis of classification algorithms for intrusion detection"North Dakota State University of Agriculture and Applied Science; Master's Thesis.

Doyen Sahoo, Chenghao Liu, and Steven CH Hoi. (2019) malicious url detection using machine learning: A survey. arXiv preprint arXiv:1701.07179.

Mohd Helmy Abd Wahab, Mohd Norzali Haji Mohd, Hafizul Fahri Hanafi, Mohamad Farhan Mohamad Mohsin (2008)" Data Pre-processing on Web Server Logs for Generalized Association Rules Mining Algorithm" PROCEEDINGS OF WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY VOLUME 36 DECEMBER 2008 ISSN 2070-3740

Phillip George Efthimion, Scott Payne, and Nicholas Proferes. Supervised machine learning bot detection techniques to identify social twitter bots. SMU Data Science Review, 1(2):5, 2018.

Shailendra Rathore, Pradip Kumar Sharma, and Jong Hyuk Park. Xssclassi_er: An effcientxss attack detection approach based on machine learning classi_er on snss. JIPS, 13(4):1014{1028, 2017.

Solomon OgbomonUwagbole, William J Buchanan, and Lu Fan. Applied machine learning predictive analytics to sql injection attack detection and prevention. In 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), pages 1087{ 1090. IEEE, 2017.

Uday Trivedi and Munal Patel. A fully automated deep packet inspection veri_cation system with machine learning. In 2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), pages 1{6. IEEE, 2016.