Journal of NAMP 69, 1 (2025) 111-126

The Nigerian Association of Mathematical Physics

Journal homepage: https://nampjournals.org.ng

ENHANCED MODEL FOR INTRUSION DETECTION IN A CLOUD ENVIRONMENT

[1]Odikayor-Ogbomo.I.F., [2]Ukueje P. O., [3]Ukaoha, K. C., [4]Malasowe, B.

[1,2]Department of Computer Science. Benson Idahosa University. Benin City, Nigeria
[3]College of Science and Computing, Wigwe University, Isiokpo, Nigeria
[4]Dept. of Cyber Security, University of Delta, Agbor, Delta State.

ARTICLE INFO

Article history:
Received xxxxx
Revised xxxxx
Accepted xxxxx
Available online xxxxx

Keywords:
Intrusion,
Detection,
Internet,
Machine learning,
Computing.

ABSTRACT

The cyber security industry is rapidly growing due to the expanding size of networks, which increases the attack surface for hackers. As cyber threats become more sophisticated, defenses must also advance. This research employs the Zhang 2019 model, utilizing a Deep Generative Neural Network (DGNN) for adversarial learning. DGNN comprises two components: G-net, which generates malicious inputs (intrusion samples), and D-net, which identifies these from real inputs. Over time, G-net produces more sophisticated malicious inputs as D-net improves its detection. The developed model monitors and analyzes data in a cloud computing environment, detecting intrusions from both external and internal users using Support Vector Machine and Bayesian network techniques. Data is collected, pre-processed, normalized, and classified into normal (ND) or intrusion data (ID). The normal data is treated as regular network requests, while intrusion data undergoes further processing before entering the deep learning training model.

In the testing phase, a test data was obtained with the use of data packet receiver and classification of the intrusions as True positive, false positive, true negative and false negative. Each frame was analyzed for true positive and true negative, the result gotten from the classification and analyses of the intrusions were used to compare and evaluate the performance of the enhanced intrusion detection model to ascertain if it was better than the Zhang model. The enhanced model effectively predicts and detects intrusion attacks and unauthorized users, providing valuable information about malicious network traffic and alerting security personnel to potential invasions.

1. Introduction

The days when only strong passwords and firewalls were all that was required to secure corporate networks, have long passed. Data integrity cannot be protected from outside intruders in today's Internet environment using common mechanisms such as ordinary password and file security, the need for adequate system security is of course the first step in ensuring data protection. For example, it is pointless to attach a system directly to the Internet and hope that nobody breaks into it, if it has no administrator password, intruder attack methodology has become more targeted and sophisticated. Measures beyond those normally expected of an intranet system should always be made on any system connected to the internet.

*Corresponding author: Ukaoha, K. C
E-mail address: kingsley.ukaoha@wigweuniversity.edu.ng
https://doi.org/10.60787/jnamp.vol69no1.468

Intrusion detection system takes that one step further, a network based intrusion detection system can provide an extra layer of protection to that system. An intrusion detection system (IDS) monitors network traffic for suspicious activity and alerts network administrators, or responds by taking predefined action. An IDS system is important as it acts as an adaptable safeguard technology for system security, consistently monitoring network traffic and system activities within the data center environment. This real-time monitoring allows the IDS to detect potential security threat as they arise, providing instant alerts that can help system administrators respond quickly and mitigate any potential damage. IDS performs an analyses of passing traffic, recognize attack patterns within the network packets, identify malicious activity, and detect external/internal hackers and network-based attacks.

1.1. Support Vector Machines

A Support Vector Machine (SVM) is a supervised machine learning algorithm that can be employed for both classification and regression purposes. SVMs are based on the idea of finding a hyper plane that best divides a dataset into two classes. Hyper plane is also a line that linearly separates and classifies a set of data. The distance between the hyper plane and the nearest data point from either set is known as the margin as shown in figure 1. The goal is to choose a hyper plane with the greatest possible margin between the hyper plane and any point within the training set, giving a greater chance of new data being classified correctly.

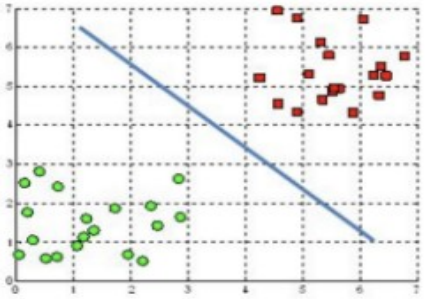

Figure 1. Hyper planes in 2D feature space. (Gandhi, 2018)

If data is linearly arranged, then we can separate it by using a straight line, as shown in figure 1, but for non-linear data, we cannot draw a single straight line.

1.2 Bayesian_network model

Bayesian networks are a type of probabilistic graphical model comprised of nodes and directed edges as shown in figure 2. Bayesian network models capture both conditionally dependent and conditionally independent relationships between random variables. Models can be prepared by experts or learned from data, then used for inference to estimate the probabilities for causal or subsequent events. Consider a problem with three random variables: A, B, and C. A is dependent upon B, and C is dependent upon B, the conditional dependencies are as follows:

i. A is conditionally dependent upon B, e.g. P(A|B)
ii. C is conditionally dependent upon B, e.g. P(C|B)

C and A have no effect on each other. Also the conditional independencies as follows:

i. A is conditionally independent from C: P(A|B, C)
ii. C is conditionally independent from A: P(C|B, A)

Notice that the conditional dependence is stated in the presence of the conditional independence. That is, A is conditionally independent of C, or A is conditionally dependent upon B in the presence of C, also the conditional independence of A given C as the conditional dependence of A given B, as A is unaffected by C and can be calculated from A given B alone. (Jason, 2019)

$$P(A|C, B) = P(A|B) \quad (1)$$

B is unaffected by A and C and has no parents; therefore the conditional independence of B from A and C can be stated as P(B, P(A|B), P(C|B)) or P(B).
The joint probability of A and C given B or conditioned on B as the product of two conditional probabilities; is:

$$P(A, C \mid B) = P(A|B) * P(C|B) \quad (2)$$

The model summarizes the joint probability of P(A, B, C), calculated as:

$$P(A, B, C) = P(A|B) * P(C|B) * P(B) \quad (3)$$

And the graph can be drawn as follows:

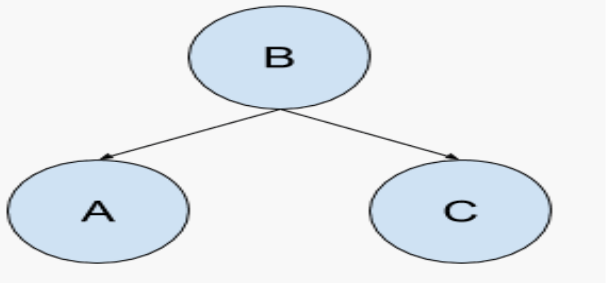

Figure 2. A simple Bayesian Network (Jason, 2019)

Note that the random variables are each assigned a node, and the conditional probabilities are stated as directed connections between the nodes, and it is not possible to navigate the graph in a cycle, e.g. no loops are possible when navigating from node to node via the edges as shown in figure 2.

2 RELATED WORKS

There are abundant literatures on intrusion detection system, and several IDS approaches have been proposed. Ayesha and Manivannan. (2021) presented a comprehensive survey of machine learning based approaches *as* presented in various literatures for the last ten years that would serve as a supplement to other general surveys on intrusion detection as well as reference to recent work done in the area for researchers working in Machine Learning-based intrusion detection systems.

Kathryn *et al,* (2021) The authors presented an extensive overview, implementation, and cross comparison of state of the art machine learning based methods available for intrusion detection, analyses some of the current state of the art intrusion detection methods and discusses their advantages and disadvantages. Machine learning algorithm using four methods was used to classify attacks to detect if traffic is benign or an attack.

Stephen *et al.* (2022) the authors presented a review of hybrid deep learning models for network intrusion detection, its concepts, characteristics and A taxonomy of deep learning approaches

was presented taking into account the deep networks for discriminative or supervised learning, generative or unsupervised learning, and finally hybrid learning that can be used to design a variety of Network intrusion detection systems.

Abdallah.E., *et al.* (2022) investigated the subject of intrusion detection using supervised machine learning algorithms methods based on a study of four popular data sets KDD'99, NSL-KDD, CICIDS2017, and UNSW-NB15 and a taxonomy for linked intrusion detection system was provided. However, data imbalance is still a major concern.

Sang-Jun-Han *et al*. (2003) proposed intrusion detection techniques by combining multiple hosts in order to detect multiple intrusions and to reduce false-positive rate. Hidden Markov Model (HMM) is a speech recognition technique that is used for modeling the system call events. Statistical technique gives the percentage of resource usages and system call events. Decision tree was used to model or classify the type intrusion to examine the future challenges. This technique has advantage of less false-positive rate that increases performance of detection. Hwang *et al*. (2007) proposed a hybrid system that combines a signature-based IDS with an anomaly detection system in a cascade structure, achieving twice the detection accuracy of IDS only system. Gómez *et al*. (2009) extended SNORT (a powerful open-source intrusion detection system (IDS) and intrusion prevention system (IPS) that provides real-time network traffic analysis and data packet logging) by adding anomaly detection preprocessor. Afterwards, various hybrid systems were proposed following the same aim, to have the strengths of both signature and anomaly based detection. Kleber and Schulter (2010) proposed a hybrid intrusion detection system for cloud and grid environment. This system cannot be deployed into a real time distributed environment, as the system cannot synchronize well with the other intrusion detection systems in the network and the architecture and working of both models is completely different. Tupakula et al (2011) have proposed a Hybrid Intrusion Detection System for Infrastructure as a Service Cloud. This system cannot handle large scale, dynamic, multithread and data processing environment. Since the system has been proposed for Infrastructure as a Service Cloud, the synchronization character is not applicable to the system. Software as a Service and Platform as a Service are the other two services of cloud, which has not been considered by the authors. Hisham *et al.* (2012) proposed a framework for intrusion detection in cloud systems. This framework can partially handle large scale, dynamic data which is another drawback of the system. The authors did not narrate the scope for implementing the algorithm for the private cloud environment. Zamani and Movahedi (2013) presented a review article on some influential algorithms based on machine learning approaches used in intrusion detection. Zamani explored that using a machine learning approach for intrusion detection enables a high detection rate and low false-positive rate with the capabilities of quick adaptation toward changing intrusive behavior. The analyzed algorithms have been categorized into artificial intelligence (AI) and computational intelligence bases. Elike *et al*. (2017) presented a taxonomy that classifies DL (Deep learning) models into generative and discriminative architectures. However, the authors also note that both CNN (Convolutional Neural Network) and DBN (Deep Neural Network) have not been exploited in the field of IDS (Intrusion Detection System) to detect attacks. Further, the authors compared the performance of DL and shallow learning models in the field of IDS. Zhang et al. (2019). Proposed a system to detect intrusion using Deep Generative Neural Network (DGNN) that performs Adversarial learning with Data Augmentation in intrusion detection. With the use of data Augmentation in intrusion detection, the detection rate and precision was better. But when there is not enough test data for the system to train with, the system experiences data scarcity and imbalance. Zeeshan *et al.* (2021) discussed the cyber security technology trends in intrusion detection utilizing ML (Machine Learning) and DL (Deep

Learning) methods. However, the present work does not cover all the methods in the intrusion detection domain; furthermore, the authors use few benchmark datasets for the model, and the analysis is not uniform. None of the work covers a deep and insightful analysis of the performance of the model. Adel et al. (2022) identified the power of various machine learning (ML) algorithms and analyzed the effect of ML algorithms for intrusion detection.

3. RESEARCH METHODOLOGY

The methodology adopted is the Object-Oriented Analysis and Design (OOADM). The object-oriented analysis focuses on the definition of classes and the manner in which they collaborate with one another to effect customer requirements. Unified modeling language (UML) and the Unified Process are predominantly features of object oriented Analysis.

The Object-Oriented Analysis and Design (OOADM) is a generic model, based on the object oriented paradigm that provides the designer with the semantics and notation necessary for the development of web based interfaces and its connections with previously existing application logic modules (Dayanand *et al.* 2012).

3.1. SYSTEM DESIGN

3.1.1. The Existing System

Deep Generative Neural Network (DGNN) are neural networks that perform adversarial learning. Adversarial learning is the process in which a machine learning model is fooled by being given spoofed input. Thus, the DGNN has two components: G-net and the D-net. Both, the G-net and the D-net are Deep Neural Networks (DNN). The G-net tries to generate malicious inputs, which in this case are intrusion samples, while the D-net tries to identify the malicious inputs from the real inputs. As the model continues to run, G-net will generate better malicious inputs as D-net gets better at identifying them (Zhang, 2019). This battle allows the DGNN to create samples that, although augmented, comes close to how real intrusions look. Once the Poisson-Gamma joint probabilistic model (PGM) generates the synthetic data, the DGNN is fed with both the synthetic and real intrusion data. The DGNN, through adversarial learning, will converge to create augmented intrusion data as shown in Figure 3, which is close to real intrusion data. This augmented data is now mixed in with normal data. The benefit of the Data Augmentation module is that now the Machine learning (ML) model, either shallow or deep, has a dataset of an equal number of normal and malicious data points (Zhang, 2019). Figure 3. Shows a model of the described Zhang model. (Zhang, 2019).

3.1.1.2. Model of The Existing System

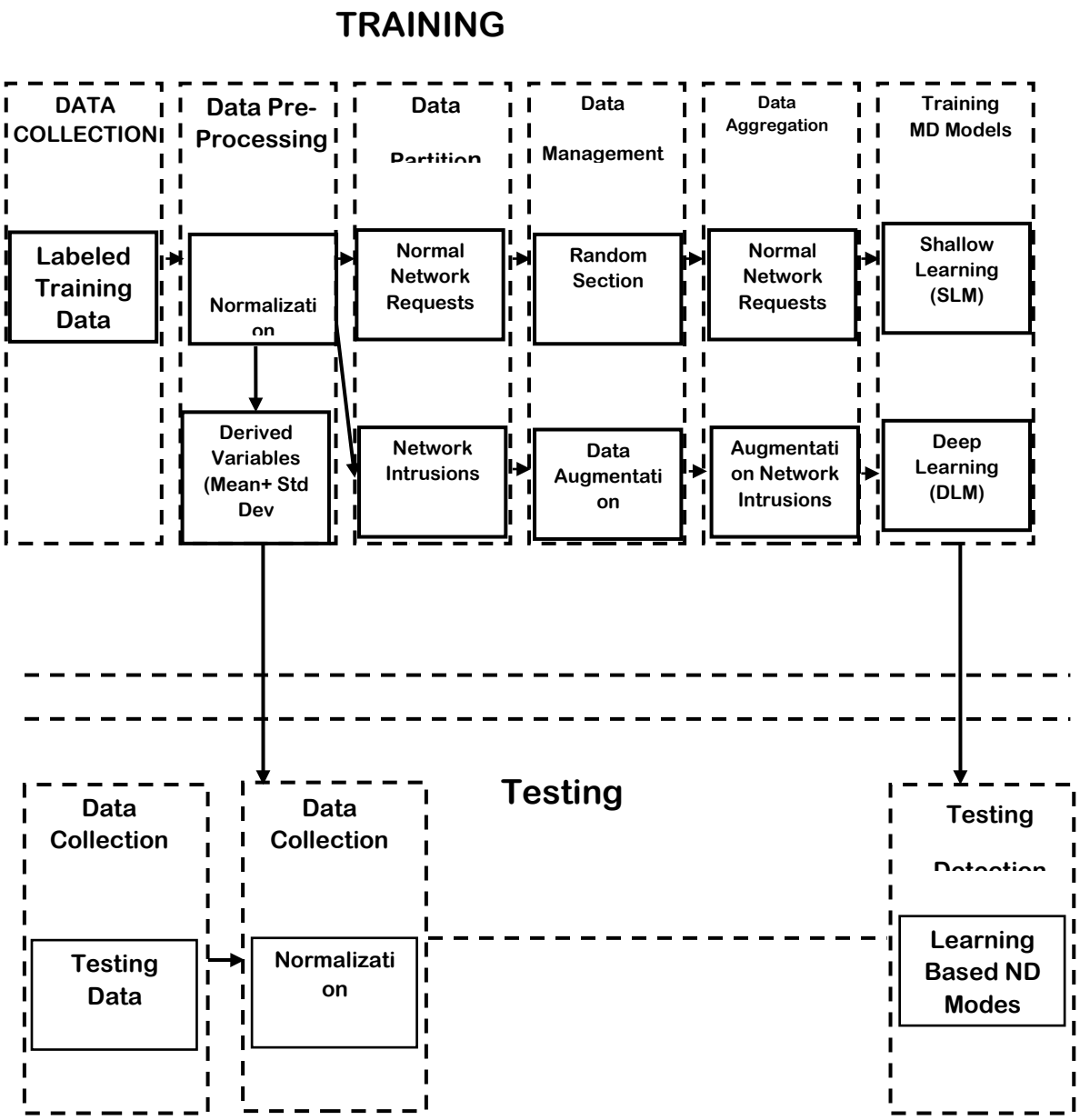

Figure 3. Network intrusion detection model using Data Augmentation (Zhang, 2019)

3.2. The New System

The new system adopted the Zhang model and enhanced it by combining two machine learning techniques (support vector machine and Bayesian network) from figure 4. Data are collected and moved to the Data pre-processing stage, where they are normalized, to reduce the height and volume of the data, and then they are classified into Normal Data (ND) or Intrusion Data (ID) using the two machine learning technique (support vector machine and Bayesian network) which was integrated after data normalization to aid in the classification of normal data and intrusion data at the data pre-processing stage. Both the Normal Data and Intrusion data are sent to the next stage which is the Data Partition where the Normal Data are now classified as Normal Network Request while the Intrusion data

are classified as Network Intrusion. The Normal Network request are sent through the Date management and Data aggregation process to the shallow learning, while the Network Intrusion (Intrusion Data) go through Data Augmentation process of Zhang (Zhang 2019) and finally to the Deep Learning Training Model. The above description constitutes the Training Phase for the models.

The testing phase include feeding the model with test data, which are then normalize and classified and a result of weather they are intrusion Data (ID) or Normal Data (ND) is made based on the result from the shallow learning process or Deep learning process of the trained model.

3.2.1 Model of the new system

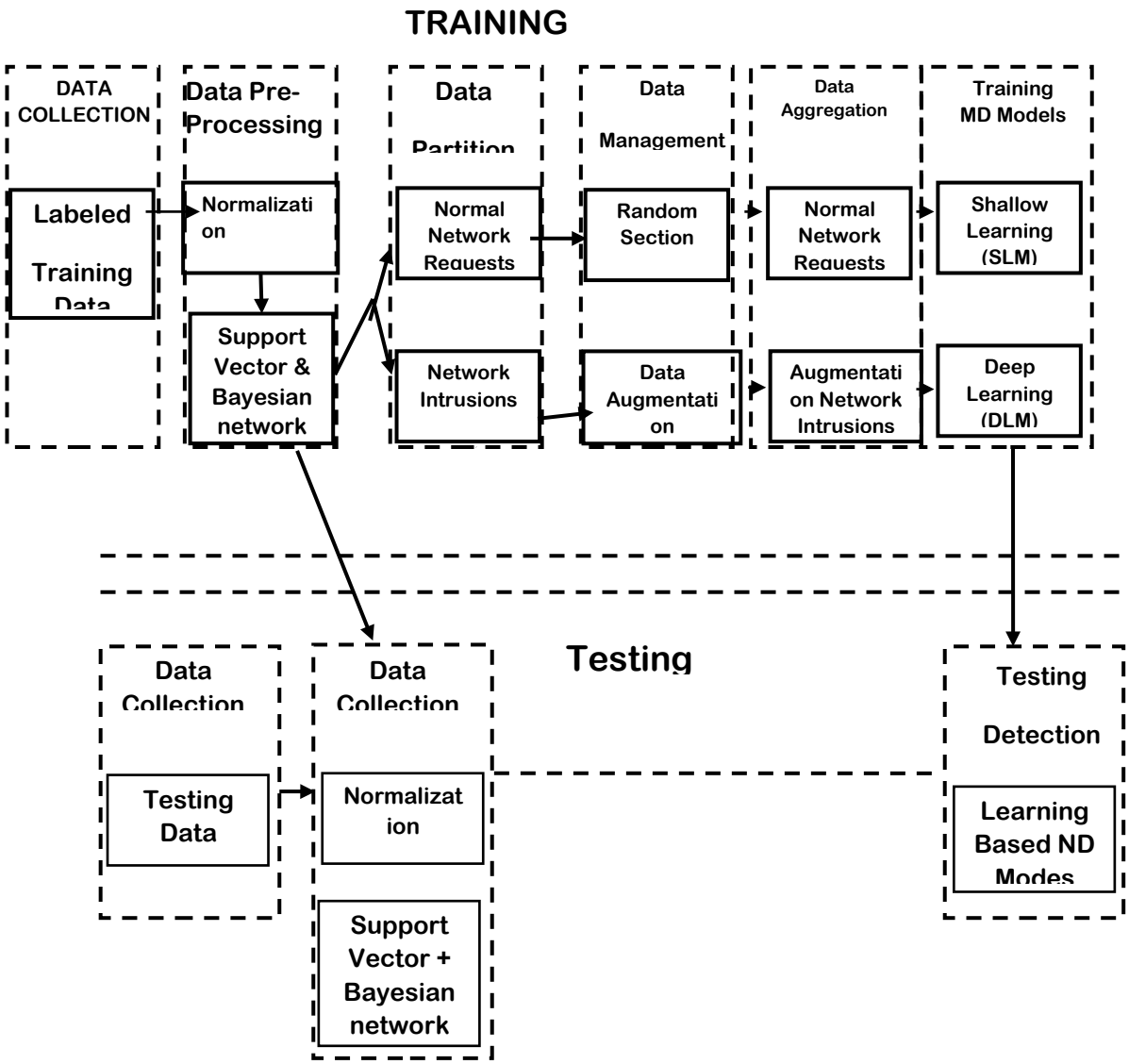

Figure 4. Model View of the New System

3.2.2. High level Model of the new System

The high level model explains the architecture that would be used for developing the automated system. The high level diagram shown in figure 5, provides an overview of the entire system, identifying the main components that would be developed and their interfaces.

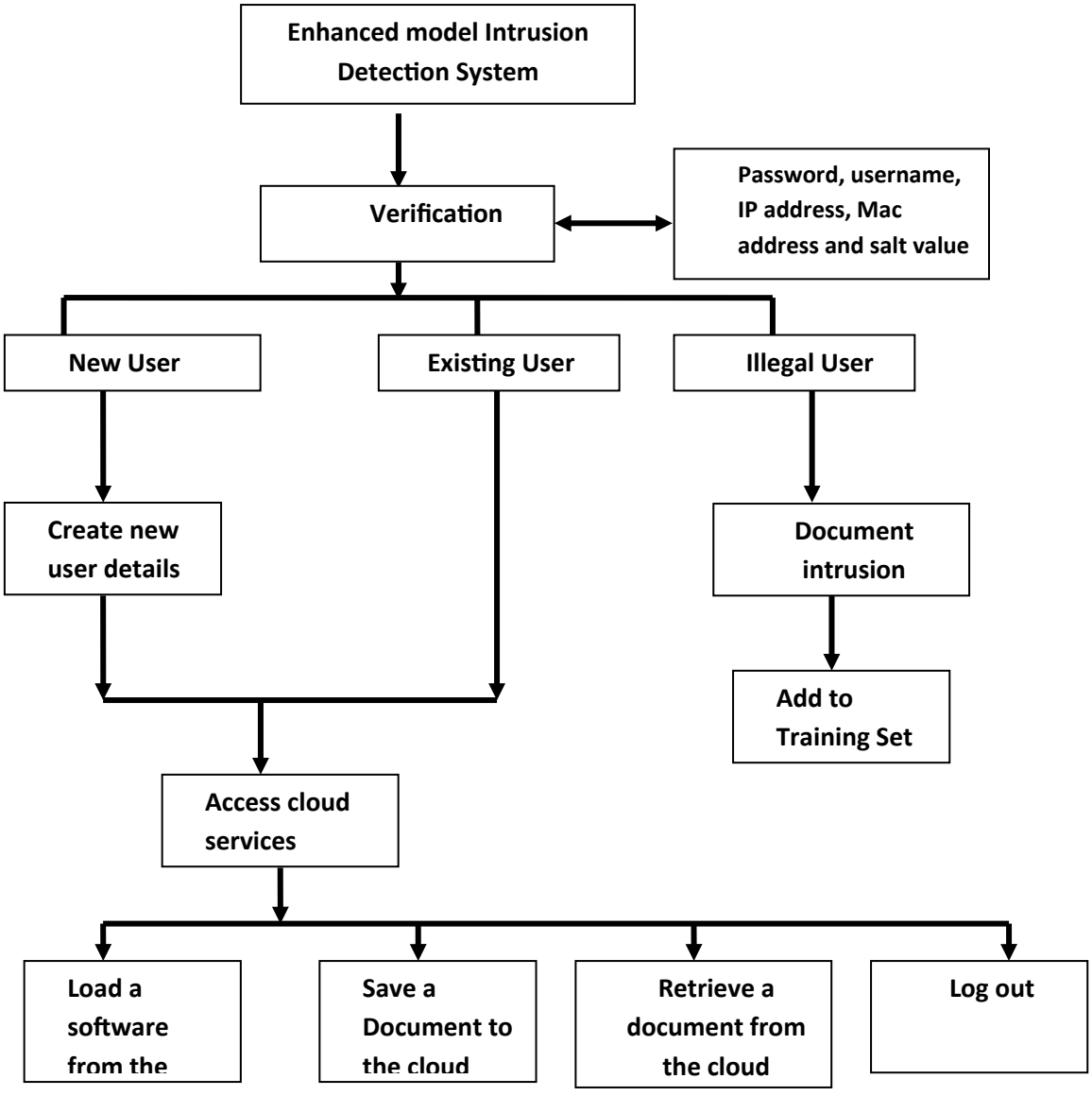

Figure 5. High level model of the new system

3.2.3. Data Flow Diagram of the New System

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows plus short text labels to show data inputs, outputs, storage points and the routes between each destination. New/existing user called web user logs into the system. If they are successful or not, the information is stored in the database server, intrusion Data and Normal Data are classified using the enhanced model and also stored in the database. Administrator controls the overall process of the intrusion detection system.

Alerts are sent to the administrator if an unsuccessful attempt to login are made as shown in figure 6.

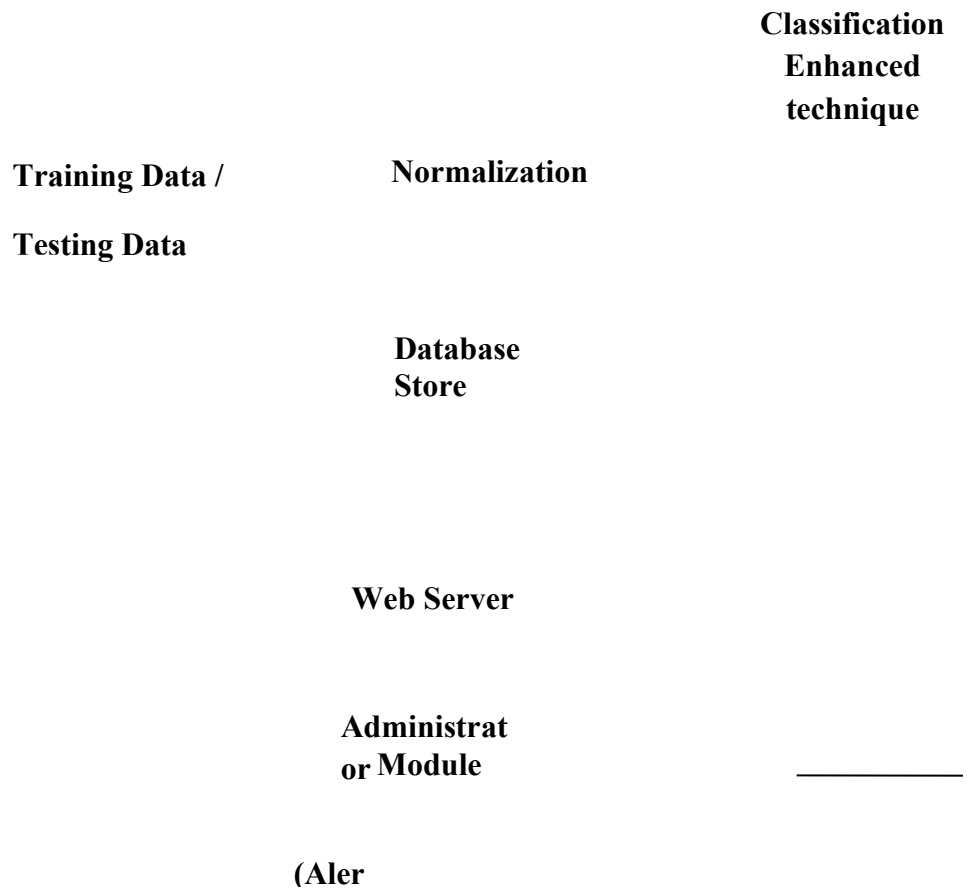

Figure 6. Data Flow Diagram of the New System.

3.2.4. Case Diagram of the new system.

Case diagram provide a model for the flow of events when it comes to user interaction. It shows a graphical depiction of a user's interaction with the system and the different types of users the system has and how users will perform tasks. It outlines from the user point of view, a system's behaviour as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user's goal and ending when that goal is fulfilled. The use case is represented by either circles or ellipses. The components of use case diagram are: Actors: any human or external system that interacts with the system. System: the main components of the system being modelled Relationships: shows how the actors and systems interact with each other. Figure 7 shows the actors (Administrators/users) interacting with the system which are the main components being modelled.

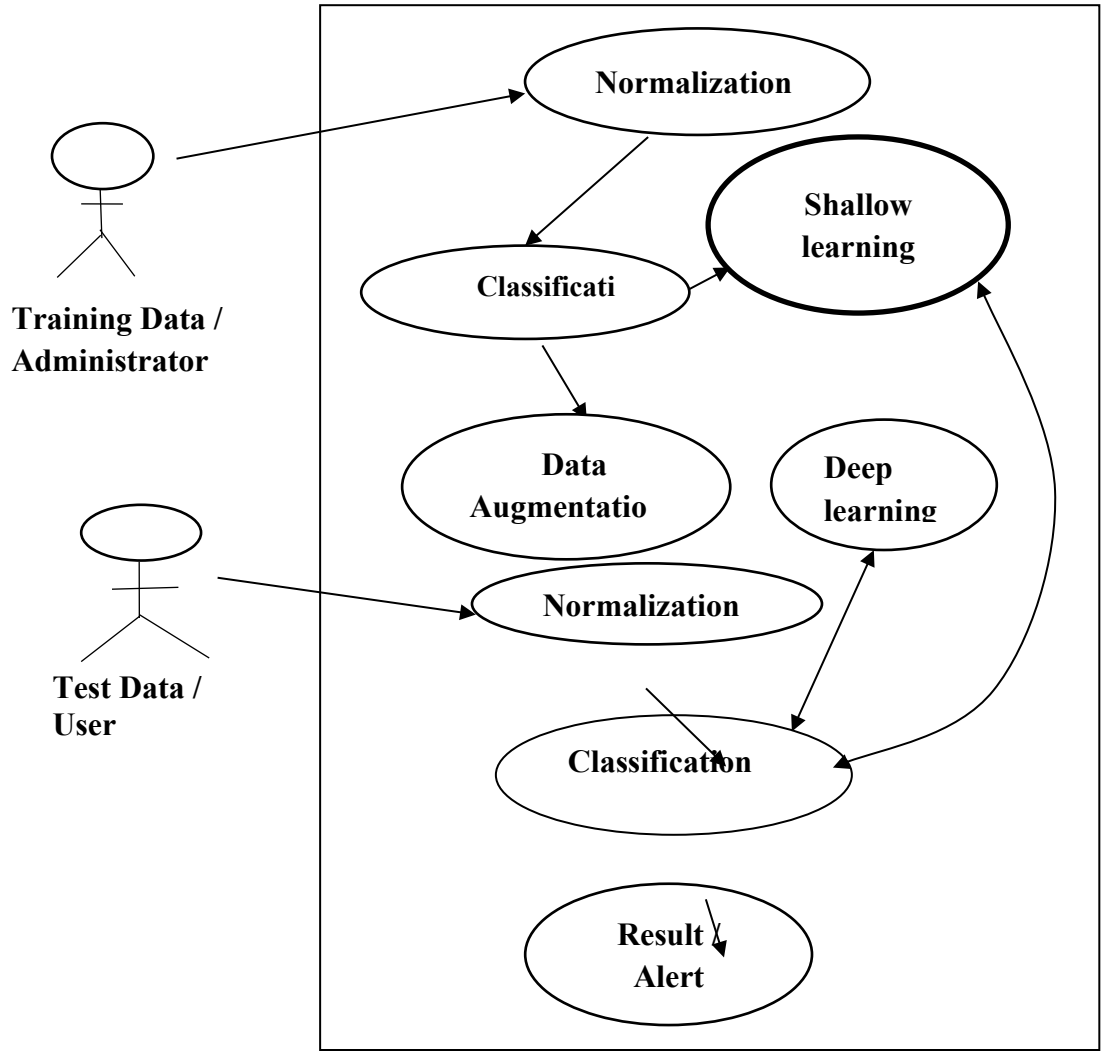

Figure 7. Case diagram of the new system.

3.2.5. System Flowchart

The system flowchart shows the flow of data to and through the major components of the system such as data entry, program storage, media, processor, and communication network. From figure 8, the user as well as the administrator can log into the system with their password. If the user is successful after his/her password has been authenticated, the user is granted access to the cloud. The administrator can select the IDS as well as test IDS. Both the user and the administrator's information are stored in the database and the result displayed.

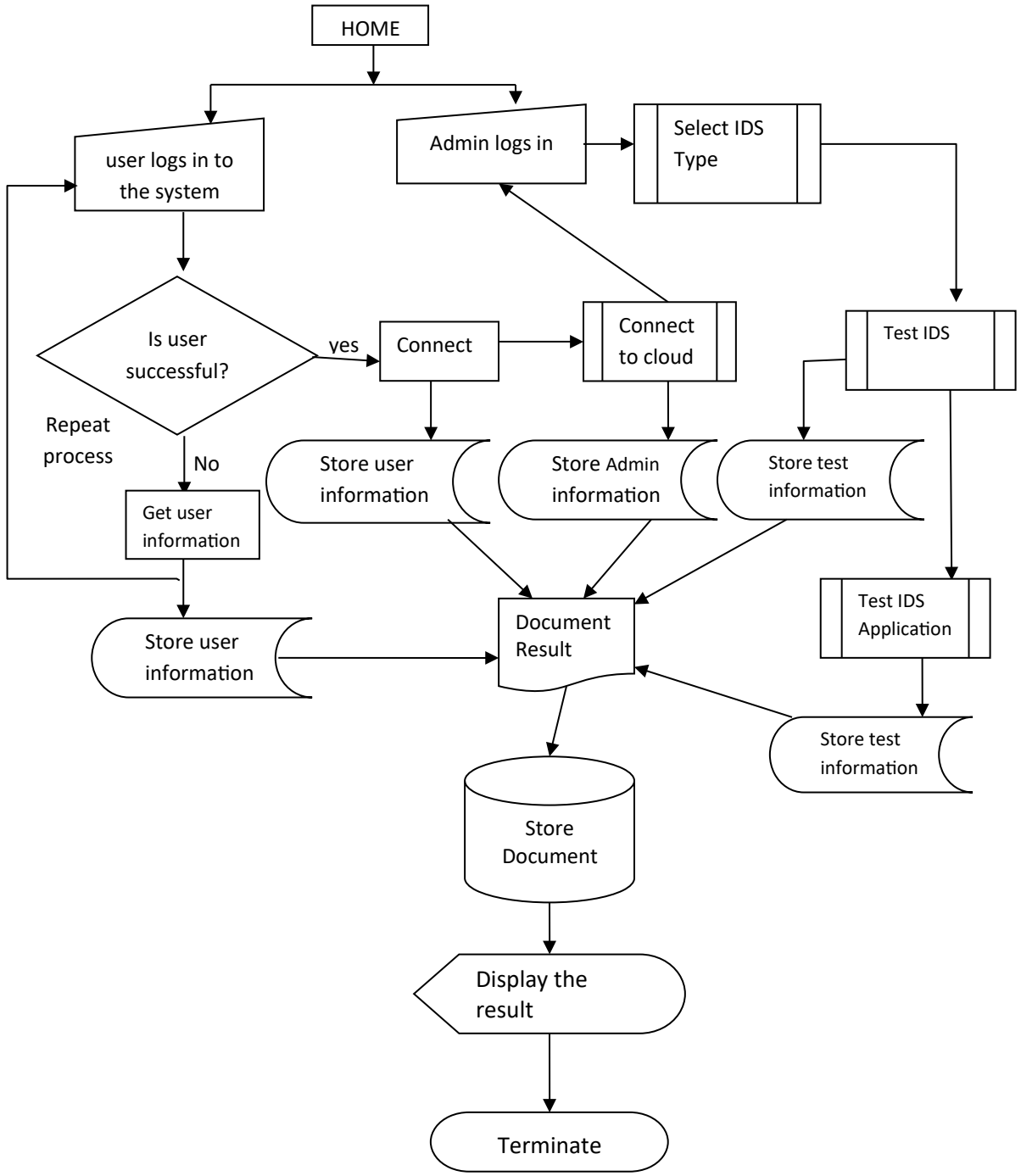

Figure 8. System Flow Chart for the New System

4. RESULTS AND DISCUSSION

The model adopted the Zhang model and enhanced it by combining two machine learning techniques (support vector machine and Bayesian network) to aid in the classification of normal data and intrusion data at the data pre-processing stage during normalization of data as shown in figure 4 to enhance the Zhang model.

A test data was obtained with the use of data packet receiver and classification of the intrusions as True positive, false positive, true negative and false negative as shown in table 1. Each frame

was analyzed for true positive and true negative as shown in table 2, the result gotten from the classification and analyses of the intrusions were used to compare and evaluate the performance of the enhanced intrusion detection model to ascertain if it was better than the Zhang model as shown in table 3. The True positive rate and false positive rate as shown in table 4 was gotten from table 3 using Excel's built-in functions. From table 3 and 4. It shows that the new enhanced intrusion detection system is more reliable than that of Zhang, as shown in the high rate of true positive and true negative value.

TRUE POSITVE **Reality:** An intrusion Attack occurs **Enhanced IDS:** Detects an Attack occurs **Output:** Record the attack as TP	**FALSE POSITVE** **Reality:** No intrusion Attack occurs **Enhanced IDS:** Detects an Attack occurs **Output:** Record the attack as FP
FALSE NEGATIVE **Reality:** An intrusion Attack occurs **Enhanced IDS:** Does not Detects an Attack	**TRUE NEGATIVE** **Reality:** No intrusion Attack occurs **Enhanced IDS:** Does not Detects an Attack **Output:** Record the attack as TN

Table 1. Classification of intrusions used to test for true positive and true negative

ID	frame	byte	Result
1	Frame 1:	208	2.05807365439093
2	Frame 2:	42	1
3	Frame 3:	42	1
4	Frame 4:	208	2.05807365439093
5	Frame 5:	208	2.05807365439093
6	Frame 6:	208	2.05807365439093
7	Frame 7:	208	2.05807365439093
8	Frame 8:	208	2.05807365439093
9	Frame 9:	208	2.05807365439093
10	Frame 10:	208	2.05807365439093
11	Frame 11:	208	2.05807365439093
12	Frame 12:	92	1.31869688385269
13	Frame 13:	92	1.31869688385269
14	Frame 14:	208	2.05807365439093

ID	frame	byte	Result
15	Frame 15:	92	1.31869688385269
16	Frame 16:	80	1.24220963172805
17	Frame 17:	158	1.73937677053824
18	Frame 18:	66	1.15297450424929
19	Frame 19:	66	1.15297450424929
20	Frame 20:	54	1.07648725212465
21	Frame 21:	54	1.07648725212465
22	Frame 22:	72	1.19121813031161

Table 2. Analysis of each frame for true positive and true negative

The result from table 2, was used to test for true positive and true negative from which the performance evaluation was done for the new enhanced intrusion detection system as shown in table 3.

4.1. Performance Evaluation

Table 3. Performance evaluation carried out on the new system

Number of Packet Frame Analyzed	Number of True Positive	Number of True Negative
12362	1908	10454
6120	954	5166
12240	1908	10332
6006	945	5061
5945	945	5000
5823	943	4880
11829	1888	9941

4.1.2. Analysis presenting TPR (True Positive rate) and FPR (False Positive rate).

Row Labels	Grand Total	TPR=TP/(TP+FN)	FPR=FP/(FP+TN)
4880	943	0.161944	0.838056
5000	945	0.158957	0.841043
5061	945	0.157343	0.842657
5166	954	0.155882	0.844118
9941	1888	0.159608	0.840392
10332	1908	0.155882	0.844118
10454	1908	0.154344	0.845656

Grand Total	**9491**		

Table 4. Showing True positive rate (TPR) and False positive rate (FPR)

4.2. Receiver Operating Characteristic (Roc) Curve

The ROC (Receiver Operating Characteristic) curve of the system is a graphical representation of the performance of the binary classification model. It plots the True positive Rate (Sensitivity) against the False Positive Rate (1- Specificity).

ROC curve helps to evaluate the model's ability to distinguish between positive and negative classes and compare the performance of the different models. It also evaluates the performance of predictive models and classification algorithms. As shown in Table 5 and figure 4. The ROC curve was created using Excel's built-in functions.

True positive rate (TPR)	False Positive rate (FPR)
0.161944	0.845656
0.159608	0.844118
0.158957	0.844118
0.157343	0.842657
0.155882	0.841043
0.155882	0.840392
0.154344	0.838056

Table 5. TPR and FPR values

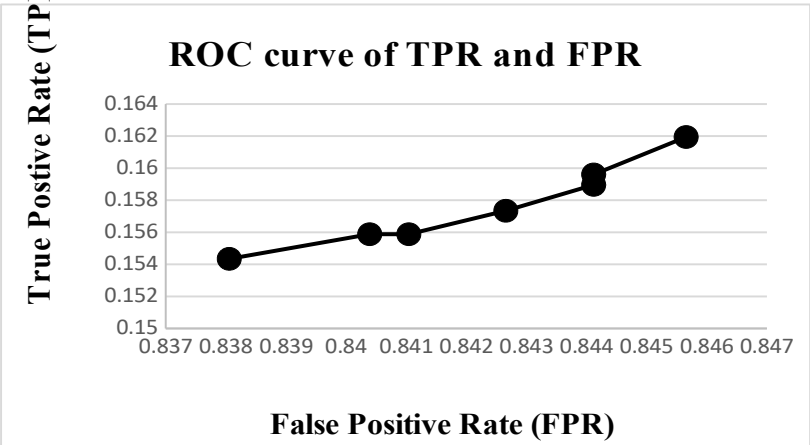

Figure 4. The ROC curve of TPR and FPR

4.3 CONCLUSION

Networks security problems vary widely and can affect different security requirements including authentication, integrity, authorization, and availability. Intruders can cause different types of attacks on systems in an organization. These attacks need to be detected as soon as

possible to prevent further damages to organizations sensitive data which may cause financial loss.

This research adopted the Zhang model (Zhang *et al*, 2019) and enhanced it by combining the model with two machine learning techniques (support vector machine and Bayesian network) to aid in the classification of normal and intrusion data during normalization at the pre-processing stage of the training phase of the model. By this process improved on the Zhang model in terms of providing more accurate and more efficient intrusion detection and reducing false-positive rate. It also provided worthwhile information about malicious network traffic; helping to identify intruders and alerting system personnel (administrators) that a network invasion may be in progress. Further research work should be carried out using public cloud and WAN (Wide area network)

REFERENCES

1. Alessandri.D. (2000): Using rule-based activity descriptions to evaluate intrusion-detection systems. In *RAID2000*, H. Debar, L. Me, and S. F. Wu, Eds. Springer-Verlag, New York, NY, 183–196.

2. Adel Binbusayyis, Haya Alaskar, Thavavel Vaiyapuri, M. Dinesh. (2022) An investigation and comparison of ML approaches for intrusion detection in IoMT network J. Supercomput., 78, pp. 17403-17422

3. Ayesha S.Dina, Manivannan D. (2021) Intrusion detection based on Machine Learning techniques in computer networks. Elsevier (iot) Volume 16, December 2021, 100462

4.Kathryn-Ann Tait, Jan Sher Khan, Fehaid Alqahtani , Awais Aziz Shah , Fadia Ali Khan, Mujeeb Ur Rehman , Wadii Boulila, Jawad Ahmad1(2021) Intrusion Detection using Machine Learning Techniques: An Experimental Comparison.(Journal of information Security and Applications, 2021)

5. Stephen Kahara Wanjau, Geoffrey Mariga Wambugu, Aaron Mogeni Oirere (2022) Network Intrusion Detection Systems: A Systematic Literature Review of Hybrid Deep Learning Approaches. International Journal of Emerging Science and Engineering (IJESE) ISSN: 2319–6378 (Online), Volume-10 Issue-7, June 2022

6. Abdallah.E.Emad, Wafa Eleisah, Ahmed Fawzi Otoom. (2022) Intrusion Detection Systems using Supervised Machine Learning Techniques: A survey. Procedia computer science (volume 201, 2022, pages 205-212. Elsevier.

7. Dayanand Ingle1 and Dr. B.B. Meshram (2012): Hybrid Analysis and Design Model for Building Web Information System. IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 4, No 3, July 2012 ISSN (Online): 1694-0814.www.IJCSI.org

8. Elike Hodo, Xavier Bellekens, Andrew Hamilton, Christos Tachtatzis, Robert Atkinson. (2017) Shallow and deep networks intrusion detection system: a taxonomy and survey, arXiv preprint arXiv: 1701.02145 (2017).1-43.

9. Gupta. S, Horrow. S and Sardana. A. (2012):"A Hybrid Intrusion Detection Architecture for Defense against DDoS Attacks in Cloud Environment," Contemporary Computing Communications in Computer and Information Science, Vol. 306, ISBN: 978-3-642-32129-0, pp. 498-499.

10. Gómez J., Gil C., Padilla N., Baños R. and Jiménez C. (2009): "Design of a snort-based hybrid intrusion detection system," in Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living, pp. 515–522, Springer, Berlin, Germany, View at Google Scholar.

11. Hwang, K., Cai. M, Chen Y., Member. S, and Qin. M (2007): "Hybrid Intrusion Detection with Weighted Signature Generation over Anomalous Internet Episodes", *IEEE Transactions on Dependable and Secure Computing*, 4(1), pp. 1-15.

12. Hisham A. Kholidy, Baiardi F. (2012): CIDS: A framework for intrusion detection in cloud systems. Proceedings of 9th IEEE International Conference on Information Technology-New Generations.p. 379–85.

13. Internet-Computer-Security (2011): IPS (Intrusion Prevention System) and IDS (Intrusion Detection Systems) http://www.internet-computer-security.com/Firewall/IPS.htmlInternational Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2, March 2012 113.

14. Kleber and schulter (2010): "Intrusion Detection for Grid and Cloud computing", IEEE Journal: IT Professional.

15. McHugh J., Allen, J., A. Christie, W. Fithen, J. Pickel, E. Stoner. (2000): State of the practice of intrusion detection technologies. Technical Report CMU/SEI-99-TR-028 ESC-99-028, Pittsburgh, PA.

16. Sang-Jun Han and Sung-Bae Cho. (2003): *Combining Multiple Host-Based Detectors Using Decision Tree*, Australian Joint Artificial Intelligence Conference, (AUSAI).

17. Tupakula U, Varadharaja V, Akku N. (2011): Intrusion detection techniques for infrastructure as a service cloud. Proceedings of 9th IEEE International Conference on Dependable, Autonomic and Secure Computing. p. 744–51.

18. Zamani. M and Movahedi. M. (2013) "Machine learning techniques for intrusion detection", *arXiv: 1312.2177*.

19. Zeeshan Ahmad, AdnanShahid Khan, Wai hiang Cheah, Johari Abdullah, Farhan Ahmad. (2021) Network intrusion detection system: a systeatic study of ML and DL approaches

20. Transactions on Emerging Telecommunications Technologies, 32, Article e4150

21. Zhang He, Xingrui Yu, Peng Ren, Chumbo Luo, Geyong Min (2019). Deep Adversial learning in intrusion detection. A data Augmentation Ehanced Framework.ArXiv: 1901.07949v3 [CS.CR]