SURVEY PAPER

AN ARCHITECTURAL FRAMEWORK FOR ANT LION OPTIMIZATION-BASED FEATURE SELECTION TECHNIQUE FOR CLOUD INTRUSION DETECTION SYSTEM USING BAYESIAN CLASSIFIER

By

HARUNA ATABO CHRISTOPHER * JIMOH YAKUBU ** SHAFI'I MUHAMMAD ABDULHAMID ***

ABDULMALIK D. MOHAMMED ****

*-** PG Scholar, Department of Computer Science, Federal University of Technology, Minna, Nigeria.
*** Senior Lecturer and Head, Department of Cyber Security Science, Federal University of Technology, Minna, Nigeria.
**** Research Scholar, University of Manchester, United Kingdom.

Date Received: 11/01/2019 Date Revised: 06/02/2019 Date Accepted: 18/03/2019

ABSTRACT

Cloud computing has become popular due to its numerous advantages, which include high scalability, flexibility, and low operational cost. It is a technology that gives access to shared pool of resources and services on pay per use and at minimum management effort over the internet. Because of its distributed nature, security has become a great concern to both cloud service provider and cloud users. That is why Cloud Intrusion Detection System (CIDS) has been widely used to the cloud computing setting, which detects and in some cases prevents intrusion. In this paper, the authors have proposed a conceptual framework that detects intrusion attacks within the cloud environment using Ant Lion Optimization (ALO) algorithm for feature selection and Bayesian Classifier. This framework is expected to detect cloud intrusion accurately at low computational cost and reduce false alert rate.

Keywords: Ant Lion Optimization, Bayesian Classifier, CIDS, Feature Selection, Cloud Computing.

INTRODUCTION

Cloud computing is becoming more popular among people because of its numerous advantages, such as scalability, flexibility, and low operational cost. It is an Information Technology (IT) paradigm that helps ubiquitous access to common pools of computing resources and services that can be quickly provided with minimal management effort, through the Internet. Cloud services rendered to cloud users are of three types (Mehmood, Shibli, Kanwal, & Masood, 2015; Latiff, 2017); Software-as-a-Service (SaaS), Infrastructure-as-a-Service (IaaS), and Platform-as-a-Service (PaaS). Applications are made available to user through the internet in SaaS. Examples are email management, and Google Docs. In IaaS, customers are allowed to have access to the entire Virtual Machine. Examples are the Amazon Web Services (AWS) and Amazon EC2. Finally, PaaS offers tools for development, deployment, and to run applications.

The main objective of cloud service provider is to effectively and efficiently utilize resources within the limit of Service Level Agreement (SLA) (Madni, Latiff, & Coulibaly, 2016). Resources on cloud are provisioned via internet for scientific and the use of resources at low cost. It provides a great reduction in the cost of installing and maintaining computing resources.

The cloud has become vulnerable to attacks both from outside and inside because of its distributed nature and enterprise is worried about the safety of their resources. Insider attack is an attacker within the cloud network who tries to gain access to cloud user's resources. They could be from the cloud provider side or cloud provider itself. External attack is an attacker outside the cloud network. An attacker who is able to perform attacks through DoS/DDoS attack, phishing attacks, etc. Attackers tend to explore the distributed nature of cloud to launch attack, which affects integrity, confidentiality, and availability of services rendered by cloud (Mahajan & Peddoju, 2017). Intrusion Detection System (IDS) is been installed in cloud

computing environment to address the various attacks in cloud. It is defined as "the process of monitoring the events occurring in a computer system or network and analyzing them for signals of intrusion, which attempts to protect confidentiality, integrity, availability, or even bypassing the security mechanisms of a computer or network" (Nagar, Nanda, He, & Tan, 2017). These IDS can either be a software or hardware placed in strategic points in the system or network which detects intrusion automatically and prevents them from further attacks.

IDS can be categorized into two based on what it protects: Host Based Intrusion Detection System (HIDS) and Network Based Intrusion Detection System (NIDS). NIDS monitors network traffic to identify malicious activities while HIDS monitors host machine for activities like pattern of system call, file access, system logs to know if there is any malicious activities. Based on detection technique, it can be categorized into three: Signature or misuse Base Detection Technique, Anomaly based Detection Technique, and Hybrid based Detection Technique. Signature technique detects intrusion by matching observed signature to attack pattern in the database. This works effectively by detecting all attacks that their signatures are in the signature database. Conversely, unknown attacks will not be detected because such attacks are not in the attack database. Continuous update of the attack database is required for it to detect new attacks. Anomaly detection technique is designed to detect intrusion by identifying all activities that deviates from the normal pattern and considers them as threats. This has overcome the shortcomings of signature based technique because it has been able to identify unknown attacks including new attacks. However, there is a high probability of generating high false alarm rate in this type of technique.

False alarm rate is the ability of a system to trigger in intrusion alert when there is no intrusion. Hybrid detection technique is the combination of misuse detection technique and anomaly detection technique.

The major contributions of this research work are chronicled as shown below:

- The authors propose a feature selection technique in CIDS using ALO algorithm.
- An architectural framework is presented for ALO and Bayesian classifier to detect CIDS.
- A summary of research datasets is also detailed for CIDS for the usage of investigators.

The goal of this research work is to put forward a conceptual framework of Ant Lion Optimization (ALO) - based feature selection technique for Cloud Intrusion Detection System (CIDS) using Bayesian Classifier.

1. Related Work

A method for detecting intrusion in cloud using genetic algorithm was proposed in (Singh, Verma, Kulshrestha, & Katiyar, 2016). This method optimizes the network path for which data is transmitted thereby increasing the speed. This makes intrusion almost impossible. The authors compared four different algorithms that have been used to optimize network path (Aho-Corasick Algorithm, Split AC Algorithm, Genetic Algorithm, and Rabin Karp Algorithm), where genetic algorithm is found to be more effective. However, this method is only a network based system that prevents intrusion within the network. Hybridizing both NIDS and HIDS will provide a complete security. While, (Pratik & Madhu, 2013) presents a data mining based CIDS called "cloud intrusion detection system for masquerade attacks (DCIDSM)". It has the following components: sensor, Extraction Translation and Loading (ETL), centralized data warehousing, automated rule generation, real-time and offline detection report and analysis, and automated alert. The CIDD dataset which is a masquerade type attack was used to test the authors' system and it was able to detect malicious activities. However, their work only covers one type of attack which is masquerading attack and does not do real time detection. Meanwhile, (Aljurayban & Emam, 2015) investigates into IDS that works with different cloud layers and to detect traffics normal among monitored cloud traffics. It is called "Layered Intrusion Detection Framework (LIDF)". LIDF uses feedforward ANN as a classification tool between normal and abnormal traffic. The proposed LIDF consist of a passive traffic capturing layer which transfers raw traffic to

the reduction layer. The reduction layer filters the traffic and then passes the output to the detection engine which will now use ANN to select malicious behavior. If there is an abnormal behavior, the detection engine will be simulated to show the existence of abnormal behavior and then alerts an administrator. LIDF when tested with real traffic was able to detect normal and abnormal instances. It shows 80% and 100% in some cases. However, the framework cannot specify in terms of abnormal behavior the type of attack or intrusion.

In (Salek & Madani, 2016), an IDS architecture that is multi-level and based on different level of risk level identified for each cloud user is proposed. The authors state that if a risk level of a user is identified, a proper IDS will be chosen and activated on the users' Virtual Machine (VM). The proposed IDS has the following agents: dispatchers agent used to identify risk level for each user, IDS manager agent does the performance and accuracy in detection rate of IDSs assigned to it, and rule-set manager, which automatically downloads from rule- set database, updated set of rules because the proposed architecture is signature based. Xen Virtual Machine Monitor (VMM) has been used to develop the simulated environment, CentOS as the operating system and snort 2.9.4.6 as the IDS. Experimental results have shown that the system has been able to decrease execution time and drop packet rate, with just a little reduction in accuracy. However, scenarios whereby an IDS is configured dynamically according to dynamic security level ought to have been considered. Also this system is not able to group together the services efficiently with the same security issues.

(Bhat, Patra, & Jena, 2013) presented a Machine Learning approach for the detection of intrusion in cloud's Virtual Machines. In their work, the machine learning approaches: Naïve Bayes and random forest performs better in detecting intrusion in cloud's Virtual Machine than the traditional and the extended Naïve Bayes method. It shows that the accuracy is high and false positive rate is low. However, this method focuses only on the Virtual Machine while other area of cloud needs to be considered for the deployment of cloud IDS. (Xing, Huang, Xu, Chung, & Khatkar, 2013) examines into an OpenFlow and Snort-based Intrusion Prevention System (IPS) called "SnorFlow" that is able to detect intrusion and provides preventive measures in cloud environment. They clearly pointed out that an IPS is preferred to the IDS for automatic action towards attackers. The authors state the challenges faced by current IPS to be latency, accuracy, and flexibility. They emphasize on the use of attack graph to choose the right countermeasures and reset the cloud's virtual networking system to prevent intrusion. However, the alert interpreter module and rule generator need to be properly optimized by an optimization algorithm so that the alerts can be correlated and the network configured without breaking down all identified vulnerable service.

(Yassin, Udzir, Muda, Abdullah, & Abdullah, 2012) presented an intrusion detection service framework that is capable of identifying activities that are malicious in cloud. The framework generates alert and notifies an administrator accordingly if there is an intrusion attempt. The authors elaborate further that Cloud-Based Intrusion Detection Service (CBIDS) is composed of three major components which are: User Data Controller (UDC), Cloud Service Controller (CSC), and Cloud Intrusion Detection Component (CIDC). The CBIDS matches information from user with signature in the database then analyze the user through the user console. However, the proposed framework is focused only on signature based technique, which cannot identify zero-day or new attacks. It is also theoretical and has not been implemented.

(Rajendran, Muthukumar, & Nagarajan, 2015) suggested a systematic approach to Hybrid intrusion detection system in private cloud. They stressed the need for HIDS considering the fact that some IDS are either based on signature technique or anomaly technique and that combination of both techniques will increase the efficiency of the IDS. Its major characteristics include: dynamic nature, self-adaptive, scalability, and efficiency. This model had been executed by means of .Net framework as front end and SQL Server as back end to store information. It has been used in Microsoft Azure cloud environment and the dynamic characteristic, scalability, and self-adaptive property is achieved.

Furthermore, the efficiency property is also achieved by detecting intrusion using both anomaly and signature technique. However, the proposed model is only for private cloud, where we have limited number of cloud users.

This framework will overcome some of the challenges faced in the existing approaches discussed above. It will first select best features using Ant Lion algorithm before the prediction between normal and abnormal behavior using Bayesian Classifier. This will not just improve detection rate, but also reduce false alarm rate as shown in Figure 1.

2. Problem Formulation

The cloud is threatened by two types of attacks: Internal and External attacks. Insider attack is an attacker within the cloud network who tries to gain access to cloud user's resources. They could be from the cloud provider side or cloud provider itself. External attack is an attacker outside the cloud network. An attacker who is able to perform attacks through DoS/DDoS attack, phishing attacks, etc.

3. Proposed Ant Lion Optimization (ALO)-Bayesian Classifier (BC) Framework

3.1 ALO Algorithm

Ant Lion Optimization (ALO) is a process, which is nature inspired (Mirjalili, 2015). The ALO system imitates the hunting style of antlions in nature. It creates a small circular pit by digging backwards into the sand, and patiently waits for its prey at the bottom. When an ant or other small insect falls into it, the hunter grabs it, pull it under the sand, and injects a special liquefying agent into its meal in order to consume it. The following are formulated as a set of condition for the whole process. The random walk of ants for every iteration is formulated in equation (1)

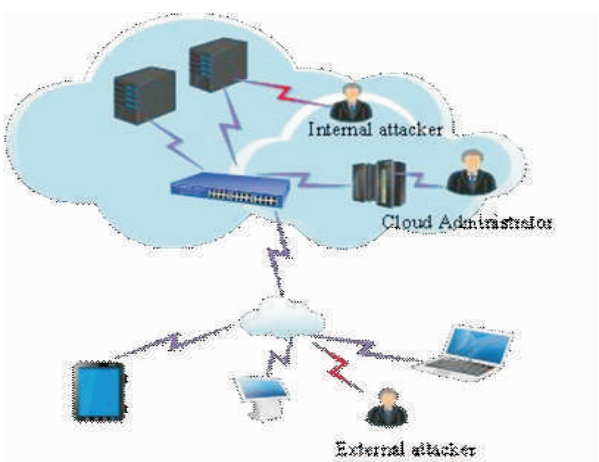

Figure 1. Cloud Attack Model

$$X(t) = [0, \text{cumsum}(2r(t_1)-1), \ldots, \text{cumsum}(2r(t_n)-1)] \quad (1)$$

where cumsum represents the accumulative total, n is the max repetition, t is the phase of random walk, and r(t) is a stochastic function which has the value (1) if a random number is less than 0.5 and 0 otherwise.

Ants use random walk to update their position at each phase during optimization. Since each exploration space has a limit, nonetheless, equation (1) cannot be openly used for apprising position of ants. For us to be able to keep within the search space random walk, it will be normalized using equation (2):

$$X_i^t = (X_i^t - a_i) * (d_i - c_i^t) / (d_i^t - a_i) + c_i \quad (2)$$

where a_i is the minimum of random walk of i^{th} variable, d_i is the maximum of random walk in i^{th} variable, c_i^t is the minimum of i^{th} variable at t^{th} iteration, and d_i^t is the maximum of i^{th} variable at t^{th} iteration.

The random walks of ants in search of a space are affected by antlions' traps modeled in equations (3) and (4).

$$c_i^t = Antlion_i^t + c^t \quad (3)$$

$$d_i^t = Antlion_i^t + c^t \quad (4)$$

where c^t is the smallest of all variables at t^{th} repetition, d^t specifies the vector with the highest of all variables at t^{th} repetition, c_i^t is the lowest of all variables for i^{th} ant, d_i^t is the maximum of all variables for i^{th} ant, and $Antlion_j^t$ indicates the position of the nominated j^{th} antlion at t^{th} repetition.

The radius of ants' random walk is reduced using equations (5) and (6) to effectively model the sliding of ant towards antlion.

$$c^t = c^t/I \quad (5)$$

$$d^t = d^t/I \quad (6)$$

When an ant is caught by the jaw of an antlion, it represents the end of hunt for the ant lion. For the authors to model this process, it is assumed that catching prey occurs when ants becomes fitter (goes inside sand) than

its corresponding antlion. An antlion is expected to increase its own chances of catching new prey by changing its position to the latest position of the hunted ant. This process is modeled below as shown in equation (7):

$$Antlion_i^t = Ant_i^t, \quad if\ f(Ant_i^t) > f(Antlion_i^t) \quad (7)$$

where t represents the current iteration, $Antlion_j^t$ shows the position of selected j^{th} antlion at t^{th} iteration, and Ant_i^t indicates the position of i^{th} ant at t^{th} iteration.

Elitism is a vital property of swarm systems that lets them to sustain the best optimal solution(s) gotten at any phase of optimization method. Since elite is the fittest antlion, it would be able to affect the activities of all the ants throughout iterations. Consequently, it is presumed that all ants walk randomly round a nominated antlion by the roulette wheel and the elite instantaneously:

$$Ant_i^t = R_A^t + R_E^t/2 \quad (8)$$

Hence, ALO algorithm is defined as (Mirjalili, 2015):

Initialize the first population of ants and antlions randomly

Calculate the fitness of ants and antlions

Find the best antlions and assume it as the elite (determined optimum)

while the end criterion is not satisfied

for every ant

Select an antlion using Roulette wheel

Update c and d using equations Eqs. (5) and (6)

Create a random walk and normalize it using Eqs. (I) and (2)

Update the position of ant using Eq. (8)

end for

Calculate the fitness of all ants

Replace an antlion with its corresponding ant if it becomes fitter Eq. (7)

Update elite if an antlion becomes fitter than the elite

end while

Return elite

3.2 Bayesian Classifier (BC)

Bayesian Classifier is an arithmetic classifier that forecasts the possibility of a given network occurrence belonging to a specific class (normal or intrusion) (Shafi'I et al., 2017; Madni, Latiff, Abdullahi, & Usman, 2017; Latiff, Madni, & Abdullahi, 2018). To calculate the probability of an event for each class is shown below:

- *Approach:*
- *compute the posterior probability P(C I A_1, A_2,….A_n) for all values of C using the Bayes theorem*

$$P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C) P(C)}{P(A_1 A_2 \ldots A_n)}$$

- *Choose value of C that maximizes P(C I A_1, A_2,…, A_n)*
- *Equivalent to choosing value of C that maximizes P(A_1, A_2, …, A_n I C) P(C)*
- *How to estimate P(A_1, A_2, …, A_n I C)?*

Let D be the be packets training dataset and their associated class tags, and vector X = (x_1, x_2, … x_n) representing each packet. If we assume that m exist as different classes C_1, C_2, .., C_m, then Bayesian classifier predicts X to be of class C_i, if the probability $P(C_i|X)$ is highest among all the $P(C_k|X)$, for all k classes. Hence, classification is derived by equation (9). Equation (10) is also maximized.

$$P(C_i|X) = \frac{P(X|Ci)P(Ci)}{P(X)} \quad (9)$$

$$P(C_i|X) = P(X|C_i)P(C_i) \quad (10)$$

3.3 ALD-BC Model

The proposed model as shown in Figure 2 consists of three major components, they are:

- *Packet capture module:* The packet capture module capture packet for proper analysis. Packet sniffing tool such as Wireshark can be used and are inspected in real time.
- *Intrusion Detection:* Intrusion detection system consists of Ant Lion optimization (ALO) and Bayesian classifier. ALO processes the network packet to remove noise that have very low correlation with detection and Bayesian classifier uses behavior base to predict class label of preprocessed packets to detect unknown attacks, storage, and alert system.
- The storage module consists of two databases; behavior base and central log. Behavior base stores

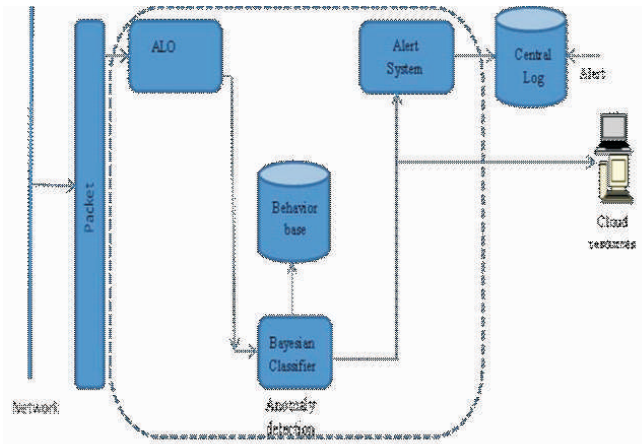

Figure 2. Architectural Framework for ALO-BC CIDS Model

behaviors of network having normal and abnormal packets. The central log stores log events of malicious activities that are considered by Bayesian classifier.

- Finally, the alert system generates alert if an abnormal behavior is noticed.

4. Summary of CIDS Datasets

This section presents a detailed summary of publicly available research dataset in CIDS. The authors have listed the reference papers, URLs, and number of instances for each dataset in Table 1.

5. Initial Results and Expected Outcome

Preliminary results of an initial experiment were obtained using BC algorithm without the ALO feature selection part. The KDD'99 dataset (CISDA, 2009; UNB, 2018) used in (Bhat et al., 2013; Idris & Abdulhamid, 2014) was utilized for the experiment. The dataset has a total of 494,021 instances. The simulation was done in a standard MATLab 7.13 R2011b data mining toolkit. Standard metrics were used for the measurement of the parametric performance as applied in (Madni, Latiff, & Coulibaly, 2017; Abdullahi & Ngadi, 2016; Latiff, Abdul-Salaam, & Madni, 2016). The results of the experiment are presented in Table 2.

- True Positive Rate (TPR) is the amount of intrusion correctly detected compared to the total number of intrusion.
- False Positive Rate (FPR) is the number of intrusion correctly generated while there is no intrusion.
- Accuracy is the percentage that the predictions made by the CIDS are true.
- Precision represents the percentage of intrusion that has occurred and CIDS detects them correctly.

Basically, CIDS has helped in detection of intrusion in cloud environment. In this conceptual framework proposed, a feature selection technique is utilized in CIDS using ALO algorithm. The authors have also presented an architectural framework for ALO and Bayesian classifier to detect CIDS. A detailed summary of research datasets for CIDS for investigators use have been outlined.

This framework is expected to improve detection accuracy, reduce detection time, and affordable computational cost.

6. Future Research

Developing a more detailed and holistic framework for Cloud Intrusion Detection system will be considered in the future. The authors will be doing a validation of the proposed conceptual framework in cloud computing environment to actually verify the expected outcome.

Conclusion

Issue of security in cloud computing has become a major concern and has reduced the rate of acceptance of

TPR	FPR	Accuracy	Precision
0.95	0.2	97.4%	98.5%

Table 2. Initial Results using BC Algorithm

Sl.No	Ref.	Dataset	Addresses	Number of instances
1	(Bhat et al., 2013; Modi& Patel, 2013)	KDD'99 NSL- KDD	(CISDA, 2009; UNB, 2018)	494,021
2	(Modi, Patel, Borisanya, Patel, & Rajarajan, 2012)	NSL-KDD KDD	(ISCX, (2007, KDD, 1999)	125,973 494,021
3	(Pratik & Madhu, 2013)	CIDD	(CIDD, n.d.)	Not disclosed
4	(Aljurayban & Ema, 2015)	ISOT	Not disclosed	Not disclosed
5	(Salek & Madani, 2016)	Darpa99	(MIT, 1999)	17634

Table 1. CIDS Datasets

cloud technology. That is why CIDS has been widely deployed in cloud to reduce the issue of cloud attacks.

In this research work, a best possible technique for cloud IDS is proposed. It uses Ant Lion Optimization technique for feature selection and Bayesian classifier for classification between normal and abnormal network traffic. It is expected to identify intrusion effectively at low computational cost with low false alarm rate.

References

[1]. Abdullahi, M., & Ngadi, M. A. (2016). Symbiotic Organism Search optimization based task scheduling in cloud computing environment. *Future Generation Computer Systems,* 56, 640-650.

[2]. Aljurayban, N. S., & Emam, A. (2015, March). Framework for cloud intrusion detection system service. In *Web Applications and Networking (WSWAN), 2015 2nd World Symposium on* (pp. 1-5). IEEE.

[3]. Bhat, A. H., Patra, S., & Jena, D. (2013). Machine learning approach for intrusion detection on cloud virtual machines. *International Journal of Application or Innovation in Engineering & Management (IJAIEM),* 2(6), 56-66.

[4]. CIDD (n.d.). *Cloud Intrusion Detection Dataset* [Dataset]. Retrieved from http://www.di.unipi.it/~hkholidy/projects/ cidd/download.html

[5]. CISDA. (2009). *A Detailed Analysis of the KDD CUP 99 Dataset* [Dataset]. Retrieved from http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, Access date: 26/03/2018

[6]. Idris, I., & Abdulhamid, S. M. (2014). An improved AIS based e-mail classification technique for spam detection. *arXiv preprint arXiv:1402.1242.*

[7]. ISCX. (2007). *Information Centre of Excellence for Tech Innovation* [Dataset]. Retrieved from http://www.iscx.ca/NSL-KDD/,Access date: 26/03/2018

[8]. KDD. (1999). *Cup 1999 Data* [Dataset]. Retrieved from http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[9]. Latiff, M. S. A. (2017). A checkpointed league championship algorithm-based cloud scheduling scheme with secure fault tolerance responsiveness. *Applied Soft Computing,* 61, 670-680.

[10]. Latiff, M. S. A., Abdul-Salaam, G., & Madni, S. H. H. (2016). Secure scientific applications scheduling technique for cloud computing environment using global league championship algorithm. *PloS One,* 11(7), e0158102.

[11]. Latiff, M. S. A., Madni, S. H. H., & Abdullahi, M. (2018). Fault tolerance aware scheduling technique for cloud computing environment using dynamic clustering algorithm. *Neural Computing and Applications,* 29(1), 279-293.

[12]. Madni, S. H. H., Latiff, M. S. A., Abdullahi, M., & Usman, M. J. (2017). Performance comparison of heuristic algorithms for task scheduling in IaaS cloud computing environment. *PloS One,* 12(5), e0176321.

[13]. Madni, S. H. H., Latiff, M. S. A., & Coulibaly, Y. (2016). Resource scheduling for Infrastructure as a Service (IaaS) in cloud computing: Challenges and opportunities. *Journal of Network and Computer Applications,* 68, 173-200.

[14]. Madni, S. H. H., Latiff, M. S. A., & Coulibaly, Y. (2017). Recent advancements in resource allocation techniques for cloud computing environment: A systematic review. *Cluster Computing,* 20(3), 2489-2533.

[15]. Mahajan, V., & Peddoju, S. K. (2017, May). Integration of network intrusion detection systems and honeypot networks for cloud security. In *Computing, Communication and Automation (ICCCA), 2017 International Conference on* (pp. 829-834). IEEE.

[16]. Mehmood, Y., Shibli, M. A., Kanwal, A., & Masood, R. (2015). Distributed intrusion detection system using mobile agents in cloud computing environment. In *Information Assurance and Cyber Security (CIACS), 2015 Conference on* (pp. 1-8). IEEE.

[17]. Modi, C. N., & Patel, D. (2013, April). A novel Hybrid-Network Intrusion Detection System (H-NIDS) in cloud computing. In *Computational Intelligence in Cyber Security (CICS), 2013 IEEE Symposium on* (pp. 23-30). IEEE.

[18]. Mirjalili, S. (2015). The Ant Lion optimizer. *Advances in Engineering Software,* 83, 80-98.

[19]. MIT. (1999). *1999 Darpa Intrusion Detection Evaluation Dataset* [Dataset]. Retrieved from https://www.ll.mit.edu/r-d/datasets/1999-darpa-intrusion-detection-evaluation-dataset

[20]. Modi, C., Patel, D., Borisanya, B., Patel, A., & Rajarajan, M. (2012, October). A novel framework for intrusion detection in cloud. In *Proceedings of the Fifth International Conference on Security of Information and Networks* (pp. 67-74). ACM.

[21]. Nagar, U., Nanda, P., He, X., & Tan, Z. T. (2017, October). A framework for data security in cloud using collaborative intrusion detection scheme. In *Proceedings of the 10th International Conference on Security of Information and Networks* (pp. 188-193). ACM.

[22]. Pratik, P. J., & Madhu, B. R. (2013, July). Data mining based CIDS: Cloud intrusion detection system for masquerade attacks [DCIDSM]. In *Computing, Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference on* (pp. 1-5). IEEE.

[23]. Rajendran, P. K., Muthukumar, B., & Nagarajan, G. (2015). Hybrid intrusion detection system for private cloud: A systematic approach. *Procedia Computer Science,* 48, 325-329.

[24]. Salek, Z., & Madani, F. M. (2016, October). Multi-level Intrusion detection system in cloud environment based on trust level. In *Computer and Knowledge Engineering (ICCKE), 2016 6th International Conference on* (pp. 94-99). IEEE.

[25]. Shafi'I, M. A., Latiff, M. S. A., Chiroma, H., Osho, O., Abdul-Salaam, G., Abubakar, A. I., & Herawan, T. (2017). A review on mobile SMS spam filtering techniques. *IEEE Access,* 5, 15650-15666.

[26]. Singh, T., Verma, S., Kulshrestha, V., & Katiyar, S. (2016, March). Intrusion detection system using genetic algorithm for cloud. In *Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies* (p. 115). ACM.

[27]. UNB. (2018). *Canadian Institute for Cyber security* [Dataset]. Retrieved from http://nsl.cs.unb.ca/NSL-KDD Access date: 26/03/2018

[28]. Xing, T., Huang, D., Xu, L., Chung, C. J., & Khatkar, P. (2013, March). Snortflow: A openflow-based intrusion prevention system in cloud environment. In *Research and Educational Experiment Workshop (GREE), 2013 Second GENI* (pp. 89-92). IEEE.

[29]. Yassin, W., Udzir, N. I., Muda, Z., Abdullah, A., & Abdullah, M. T. (2012, June). A cloud-based intrusion detection service framework. In *Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2012 International Conference on* (pp. 213-218). IEEE.

ABOUT THE AUTHORS

Haruna Atabo Christopher is currently a Master Degree Scholar in Computer Science at Federal University of Technology (FUT), Minna, Nigeria. He is a Graduate in Computer Science from the same Institution. His current research interests are in Cyber Security, Cloud Computing, Soft Computing, and Big Data.

Jimoh Yakubu is currently a Master Degree Scholar in Computer Science at Federal University of Technology (FUT), Minna, Nigeria. He is a Graduate of Mathematics with Computer Science in the same institution. His current research interests are in Cyber Security, Cloud Computing, Soft Computing, and Big Data.

Dr. Shafi'i Muhammad Abdulhamid is a Senior Lecturer and Head of Department (HOD) of Cyber Security Science at Federal University of Technology (FUT), Minna, Nigeria. He is also supervising both Masters and Ph.D students (in both Nigeria and Malaysia). He received his Ph.D in Computer Science from University of Technology Malaysia (UTM), M.Sc in Computer Science from Bayero University Kano (BUK), Nigeria, and a Bachelor of Technology in Mathematics with Computer Science from the Federal University of Technology (FUT), Minna, Nigeria. He has been appointed as an Editorial board member for Big Data and Cloud Innovation (BDCI) and Journal of Computer Science and Information Technology (JCSIT). He has also been appointed as a Reviewer of several ISI and Scopus indexed International Journals. He has also served as Program Committee (PC) member in many National and International Conferences. He is one of the pioneer instructors at the Huawei Academy of FUT Minna and a holder of Huawei Certified Network Associate (HCNA). He is as well a member of IEEE Computer Society, International Association of Computer Science and Information Technology (IACSIT), Computer Professionals Registration Council of Nigeria (CPN), International Association of Engineers (IAENG), The Internet Society (ISOC), Cyber Security Experts Association of Nigeria (CSEAN), and Nigerian Computer Society (NCS). His current research interests are in Cyber Security, Cloud Computing, Soft Computing, Internet of Things Security, Malware Detection and Big Data. He has published many academic papers in reputable International Journals, Conference Proceedings, and Book chapters.

Abdulmalik Danlami Mohammed is a Ph.D Research Scholar at the University of Manchester, UK and also a member of Nigeria Computer Society (NCS) and International Association of Engineering (IAENG). He holds a B.Sc Degree in Computer Science, which was obtained from Saint Petersburg State Electro-Technical University (2003), Russia and M.Sc Degree in the same field from Belarussian National Technical University (2010), Belarus. His research interest includes Image Processing, Computer Vision, Machine learning, Mobile Computing, and Computer Networks.