International Journal of Applied Science and Mathematical Theory E- ISSN 2489-009X
P-ISSN 2695-1908, Vol. 9 No. 2 July, 2023 www.iiardjournals.org

Investigating Cloud Service Attacks using Machine Learning

Durunna Lilian Ijeoma., Agbakwuru A.O., Agbasonu V.C., Amanze B.C
Department of Computer Science, Imo state polytechnic, Omuma Oru East
Department of Computer Science, Faculty of Physical Sciences, Imo State University, Owerri
amanzebethran@yahoo.com

D.O.I: 10.56201/ijasmt.v9.no2.2023.pg66.83

Abstract
In this work, a rigorous investigation has been carried out as to the classification of cloud service attacks, utilizing machine learning algorithms. Examining the attacks emphasized that those features in the dataset are recognise as the most significant when it comes to identifying the cloud service attacks. When seeking to establish identification, cloud service attacks are viewed as being the most troublesome, predominantly considering their involvement of network- and host-level characteristics. As such, both host and network-level components-namely 'duration of connection' and 'service requested', and the 'number of failed login attempts', respectively-are selected in the establishment of cloud service attacks. In considering the form of operation manipulated by the cloud service attacks, the significant features when it comes to recognizing such forms are seen to belong to the KDD cup dataset. Importantly, most cloud applications present the potential of various logins, which fundamentally positions the end user in such a way that they are able to link to the email server through various instruments simultaneously. As such, traffic features need to be determined through the presence of a host–services connection incorporating a two-second time window. In line with the Ftp exploitation tool, which is generalized as an attack on the FTP protocol through the attacker making use of the PORT command with the aim of achieving access to ports, this is commonly recognise as an Ftp bounce attack. Accordingly, in Ftp identification, the aspects of urgent, compromised, -access-files and host-count are essential, as determined through the methods as being unique features for Ftp. In regards to the password guessing form of attack, there are also a number of different features chosen by the algorithms tested, including failed-logins, and host-error-rate. Owing to the fact that such an attack arises following various efforts being made to log in, it is possible to establish the number of failed login attempts, in addition to other features. Through a multi hop attack, it is possible for a particular attack path to be followed by an attacker, notably with the objective to achieve access to the target. As such, the most significant features recognise as unique features in the number of compromised (number of file/path not found errors) conditions, the number of shells prompts, and those that establish the Count traffic in the destination–host link, is critical. Nonetheless, there is the uploading and downloading of data from various hidden directions when there is an FTP connection, which subsequently constitutes attacks. With this in mind, the number of significant features deemed pertinent to achieving the most optimal attack outcomes is determined. Also it is possible that the attacker can gain access to the server by logging in using anonymous credentials; this, in turn,

highlights the potential for directories and files to be uploaded. In this case, it is possible to download malicious files uploaded to the server by the attacker, with any anonymous/legal user able to do so. In summarizing the whole work, the work mainly focused on cloud service attacks detection. The research work presented a system that uses machine learning to detect cloud service attack in a network and block the attack.

Keywords: *Machine Learning, attacks, cloud computing platform, Financial Institution*

INTRODUCTION

In cloud computing platforms, attackers now exploit security hole contain in deployed software applications and launch malicious attacks on target software in real-time (Dunlop et al., 2020). This prevalent malicious software attacks due to software vulnerabilities are perceived as big challenge to cloud computing acceptability (Jun et al.,2019). Many cloud customers still doubt the uncertainty and readiness to be able to guarantee and secure huge data which are to be deployed and host on the net. To resolve these security issues, a lot of techniques have been proposed including machine learning. In recent times, machine learning techniques (MLTs) have been vastly used in modelling and monitoring complex applications. Machine Learning (ML) has also shown outstanding performances in many fields including information and communication technology, by providing useful descriptive and predictive information. The main advantage of ML methods is their ability to create models that may be integrated into the decision-making process. Numerous MLTs such as Artificial Neural Network (ANN), Decision Tree (DT), Principal Component Analysis (PCA), etc. have been proposed in various capacities. Unlike traditional methods, MLTs have proven to be computationally powerful, systematic and explicitly reliable when they are deployed in classification studies. Research during the last decade has focused on developing different Machine Learning (ML) techniques for Intrusion Detection Systems (IDSs)as mentioned in (Engen, 2020). Strategies that are most commonly used are those that are able to learn from training samples illustrating typical network behaviors under different attacks. The IDSs strategically learn to detect intrusions, without the intervention of a human to identify the attack. The IDSs is able to recognize attacks after learning the typical patterns and variations seen during previous, known attacks. In previous work that used machine learning in IDS, researchers have applied known classification algorithms for detecting various types of attacks. Majority of this work has been carried out by researchers who have the computer network security expertise, but not fundamental knowledge about machine learning algorithms. Therefore, this thesis will use machine learning algorithm as a 'tool' to achieve detection of attacks in cloud computing environment. Because the internet is the delivery method of all cloud services, security challenges are often encountered. Typically, any attacks incurred by a cloud system are unique to that cloud system and can cause a lot of damages to the system. Recognizing and eliminating the attacks is critical to maintaining the confidentiality and integrity of the cloud systems and the information and resources contained in those systems. Outsider attacks by intruders are not the only threat to

cloud security. While firewalls can successfully help prevent outside intruders from attacking a cloud system, only Intrusion Detection Systems (IDSs) will be helpful in detecting insider attacks. Current IDSs are embedded with limitations, such as those related to the accuracy, sensitivity to false alarms, costs of communication, ideal detection rates, and coverage for attacks. Because of these limitations, many cloud systems are vulnerable to attacks and breaches of confidentiality. Finding solutions to these problems is critical to the integrity of any cloud system. Also, cloud services are faced with a lot of problems including:

1. Cloud services are protected using username and password to authenticate users on the platform. Password can easily be compromised and this leads to lack of adequate internet security on the cloud services platform.
2. If hackers get the security identification information of the cloud platform as a result of eave dropping or insider compromise, important data can be compromised as hackers will gain access to the network system
3. Fraudulent individuals can use fishing tools to trick users into submitting vital information to them so that access can be gained to their sensitive online accounts.
4. Attackers can send malicious code and viruses and it can slip into the network and gain access to the computer systems connected to the network causing various sorts of issues. So, the problems stated above motivated the research to conduct this research in order to proffer solutions to the existing cloud service attacks. The aim of this paper is to develop a machine learning based investigation of cloud service attacks. The objectives are to:

    1. To develop a cloud based electronic payment system for a financial institution using php-mysql and java script.
    2. To build a cloud based electronic payment system that integrate machine learning algorithm to automatically detect and classify attacks on the cloud service platform and automatically responds to the attack by blocking the malware.
    3. To determine the accuracy of detection of cloud service attacks when the system is implemented using machine learning algorithm to detect cloud service attacks.

Table1: **Summary of Related Literature**

Author	Techniques	Work done	Limitations
Folasade and Blaise (2019)	one-time password and cryptographic	Developed a brute-force prevention system for cloud computing, that protects the Symmetric key encryption algorithm	Depends on several factors, such as the lack of availability of the mobile network, which can cause a delay in obtaining the OTP
Jaspreet and Rupinder, (2015)	Genetic Algorithm	This system creates a fitness function to recognize the conceivable malicious records	Need to improve on the accuracy of the detection

Carlisle and Guy-Vincent, (2020)	Lightweight protection	This system can be utilized to slow down brute-force attacks and can likewise prevent attacks on knowledge questions	Limited to only brute-force attacks
Modi, et al. (2013)	genetic algorithms and fuzzy	Solved the best fit problem in Cloud environment	Need to improve on the accuracy of the detection
Mobin and Vern, (2018)	Aggregate Site Analyzer	Various attacks which unmistakably focused on just the local site were likewise found	users undoubtedly at times pick feeble passwords, empowering brute-forcers to infrequently succeed
Bahaa, (2012)	secure simple mail transfer protocol	The model gives remote monitoring to administrator about who attempt to hack the server through sending e-mails	Limited to DOS attacks
Nitesh, (2017)	Intrusion Detection System	It identifies faulty IPs and labels them as blacklist addresses	Cannot detect intrusions from encrypted network traffic
Satomi and Yuki, (2014)	Intrusion Detection System	watching multi servers and visualizing focused on dst-IPs and detection time	Cannot detect intrusions from encrypted network traffic

Method of Data Collection

When deploying machine learning (ML) models in the real world, anomalous data points and shifts in the data distribution are inevitable. From a cyber security perspective, these anomalies and dataset shifts are driven by both defensive and adversarial advancement. In this research work, the BPF-extended tracking honeypot (BETH) dataset as the first cybersecurity dataset for uncertainty and robustness benchmarking was used for data collection (Kate, et al 2023). The dataset has the following properties that make it attractive for the development of robust ML methods:

1. At over eight million data points, this is one of the largest cyber security datasets available
2. It contains modern host activity and attacks
3. It is fully labelled
4. It contains highly structured but heterogeneous features
5. Each host contains benign activity and at most a single attack, which is ideal for behavioural analysis and other research tasks.

The BETH dataset currently represents 8,004,918 events collected over 23 honeypots, running for about five noncontiguous hours on a major cloud provider. This subset was further divided into training, validation, and testing sets with a rough 60/20/20 split based on host, quantity of logs generated, and the activity logged—only the test set includes an attack(Kate, et al 2023). The dataset is composed of two sensor logs: kernel-level process calls and network traffic. The initial

benchmark subset only includes process logs. Each process call consists of 14 raw features and 2 hand-crafted labels.

Analysis of the Proposed System

The main goal of the proposed system is to apply a set of classification algorithms to obtain a classification model in order to be used as a scanner for cloud service attacks detection. The implementation involves tasks such as data preprocessing, feature extraction, training models etc.

In this research work, the BPF-extended tracking honeypot (BETH) dataset as the first cybersecurity dataset for uncertainty and robustness benchmarking was used for data collection. The dataset provided information about existing cloud service attacks and the characteristics. In this research, the system developed uses machine learning classifier to classify the network traffic activities. The thesis proposes a methodology to create a model that will detect if an activity on the cloud service network is normal or malicious, by applying supervised machine learning algorithms on an annotated (labeled) dataset that are classified and guaranteed.Decision Tree was used for data classifications. The decision tree is an important tool that works based on flowchart like structure that is mainly used for classification problems. Each internal node of the decision tree specifies a condition or a “test” on an attribute and the branching is done on the basis of the test conditions and result. By employing the decision tree algorithm, the research categorizes the attack by discovering the traffic of the abnormal. To address the problems of anomaly detection in the cloud environment, the research initially segregates the cloud data into two types, such as malicious and normal. Such data is fed into the multiple classifiers for acquiring individual decisions. Then, by aggregating the results of unique choices, it identifies the abnormal behaviour in the cloud infrastructure. It can detect and respond to the unknown attacks online with the minimal cost of computing. Distributed Denial of Service (DDoS) attack is the most severe attack that significantly affects the performance of the cloud environment. The use of decision trees detection techniques offer greater detection accuracy.

Data Flow Diagram (DFD) of the Proposed System

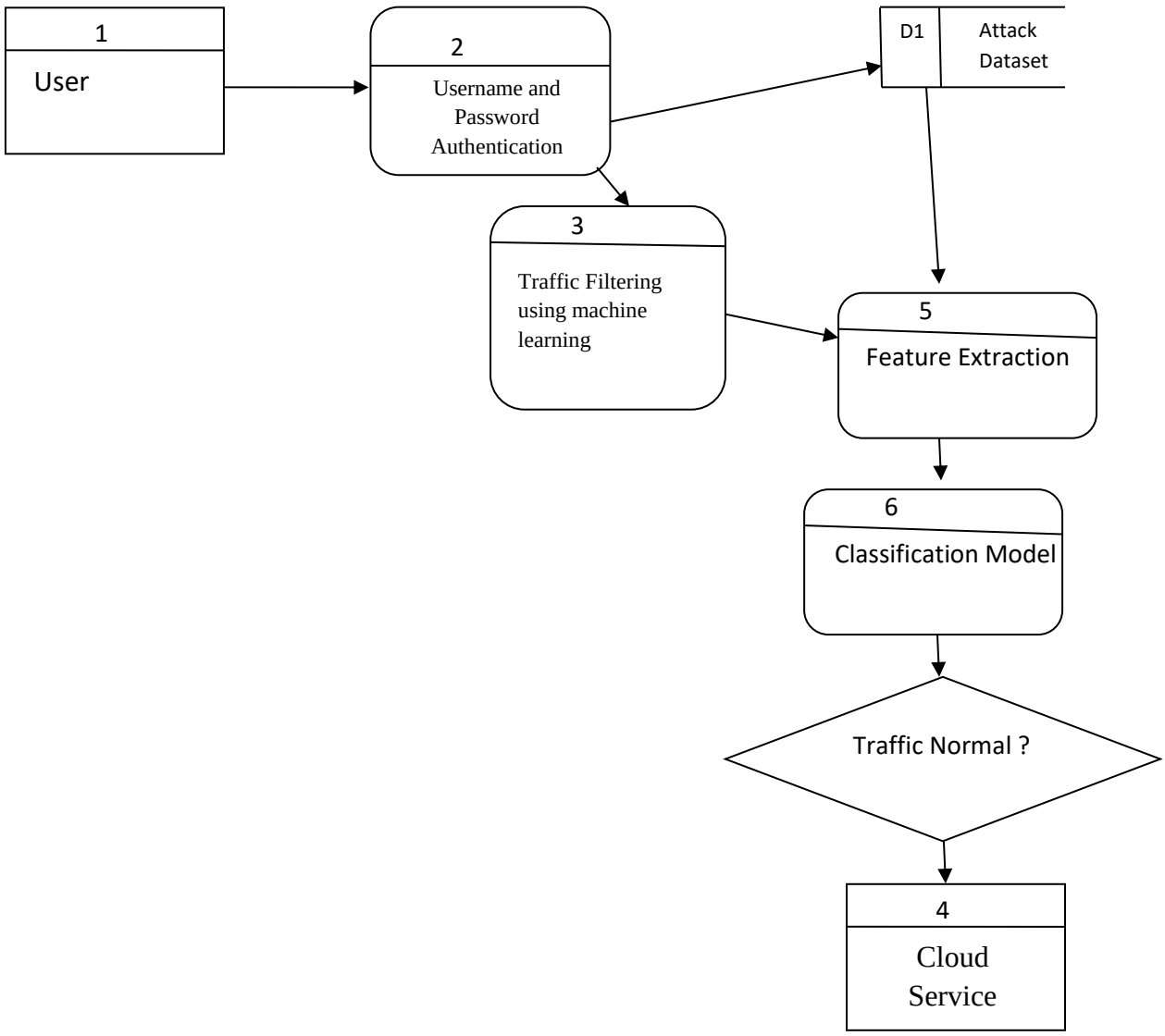

Figure 1: Data Flow Diagram of the proposed system

Use Case Diagram

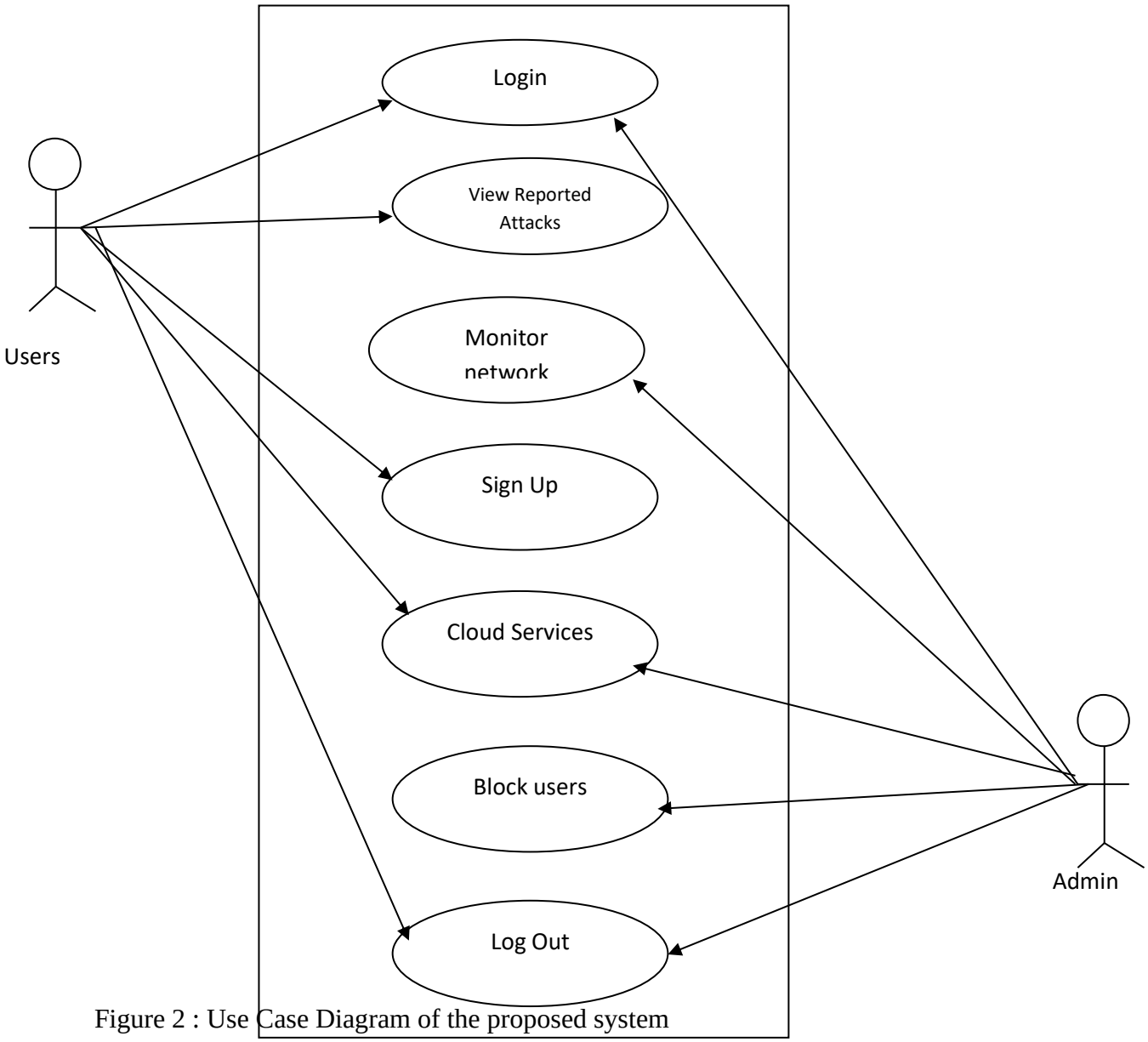

Figure 2 : Use Case Diagram of the proposed system

Sequence Diagram.

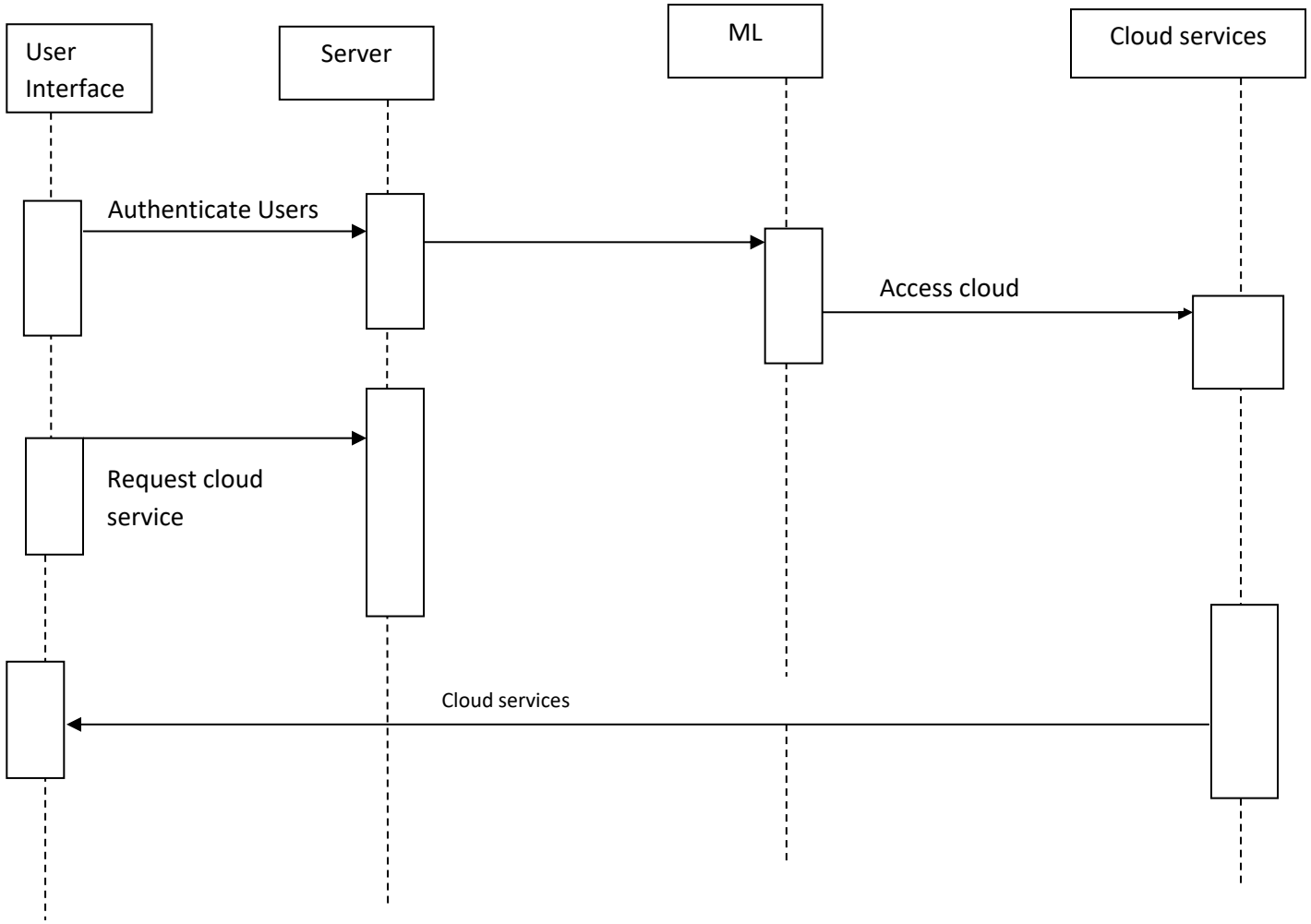

Figure 3: Sequence diagram of the proposed system

Figure 3 shows the sequence diagram for the activities of this model. The interactions include the user interface which provides graphic user interface which, interacts directly with the user, local application and server which host local application programs and data store and cloud which is located at the cloud provider side.

Activity Diagram

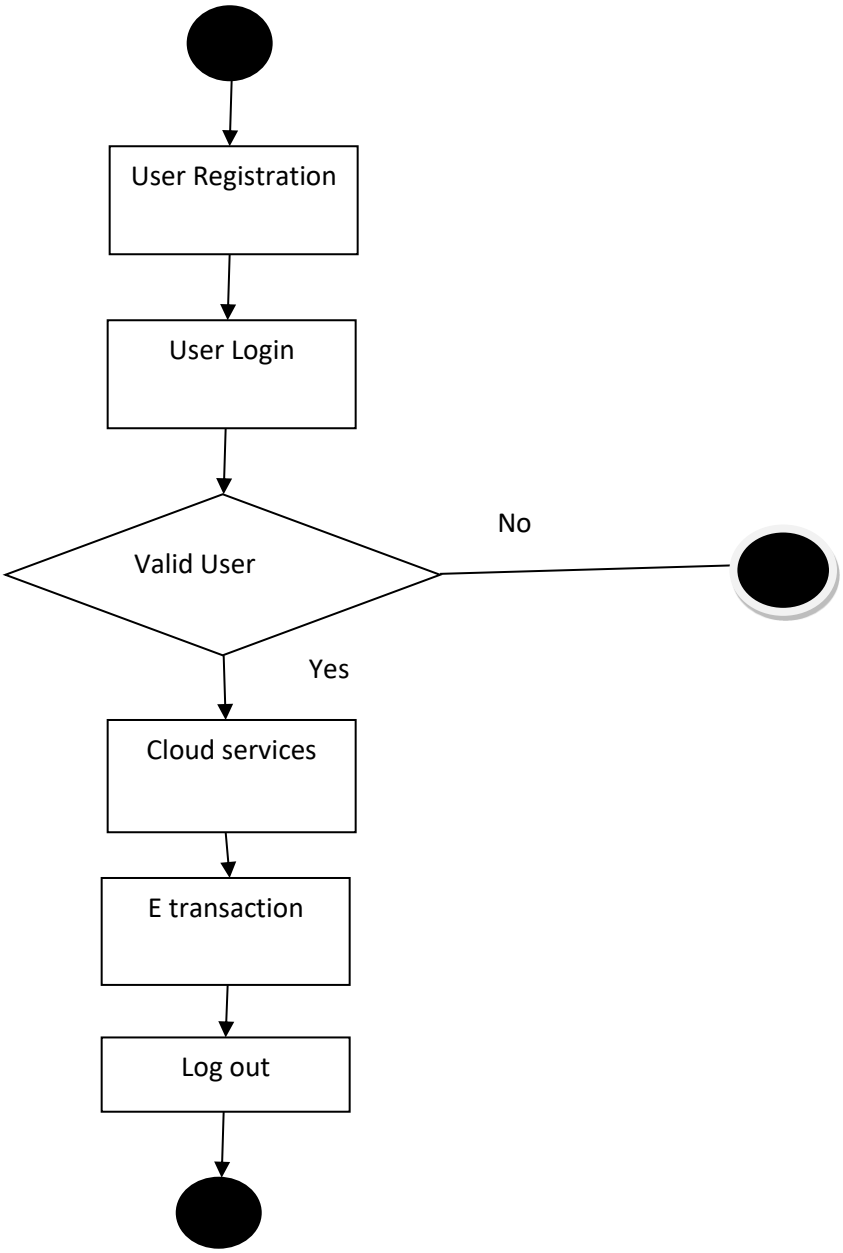

Fig.ure 4: cloud service Activity Diagram

Figure 4 shows an activity diagram for cloud services. Registration module allows a user to register with system domain. The registered user then logs in through user module. Once a user is authenticated, he receives an access key to enable user access the cloud storage servers. If user receives an access key, he accesses the storage servers and process e transactions, else activity diagram ends.

Interaction Diagram

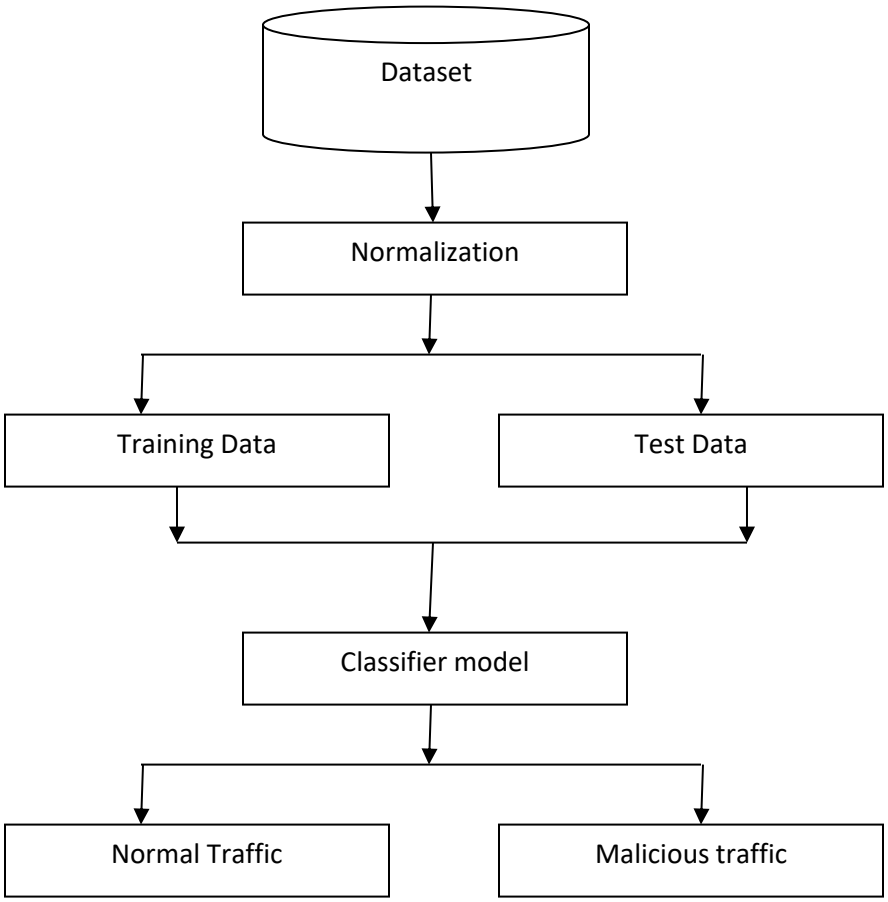

Figure 5: Interaction diagram representation of the machine learning based investigation of cloud service attack

The interaction diagram representation of the machine learning based investigation of cloud service attack is shown in figure 5. The first step in the development of the cloud service attacks

dataset construction.Following this, the training and the classifier model are used to classify the attack as normal or malicious.

High Level Model of the Proposed System

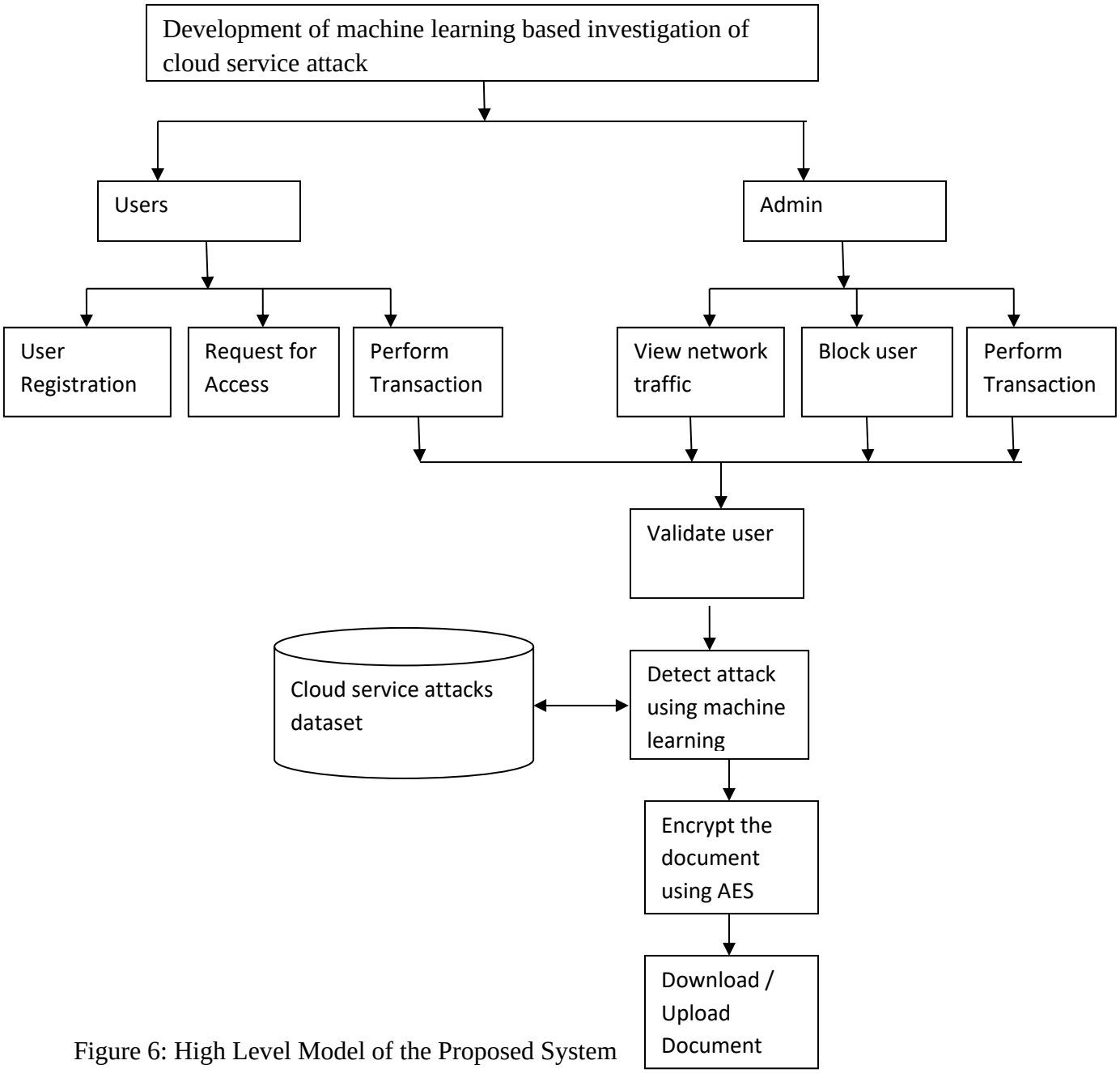

Figure 6: High Level Model of the Proposed System

Algorithm

Start,

Create an empty node,

if (T←0) →node of failure value

end if

if (T←C)→node of target attribute value

end if

if (R is empty) →node of majority attribute

end if

while (stopping criteria)

{

The attribute of highest gain ratio value is chosen node $N$is labeled with chosen attribute

For all test attribute {

Split the input sample T→$T_{1,2}$.....T_{n}

If (T_{I} is empty) {

leaf node of the majority class in input sample;

}

else {

the target value attained is attached to the leaf node

Return the tree.

The user, functional and non-functional requirements that guided the design and implementation of the proposed system includes.

Algorithm:

/* Main loop */

FOR each source Node (s) /* Concurrent activity */

Set t to be current time

WHILE t≤T /*T is the total experiment time */

Select destination node to be d;

Set T*sd* to zero /*T*sd*travel time from s to d*/

IF (G*d* =Yes)

Launch Check machine learning (s, d); /* From s to d */

ELSE

Launch supervised learning (s, d);/* From s to d */

IF(T*sd*<= *T_Goodsd*) /* extracted from T-Good table */

Set G*d*to 'yes'

END IF

END IF

END WHILE

END FOR

Launch Check machine learning (source node: s, destination node: d)

T*sd*= 0

WHILE (current_node≠ destination_node

Select next node using routing table; /* node with highest probability */

Get travel_time from the routing table of the source node;

/* from current node to next_node */

Set T*sd*to be (T*sd*+travel_time);

Set current_node to be next_node;

END WHILE

IF (T*sd*>T_Goods*d*) SetG*d*to be 'No'

CHECK cloud service attacks

Launch supervised learning (Source node: s, destination node: d)

WHILE (current_node≠ source node)

Select the next node using routing table

Push on stack (next_node, travel_time);

Set current_node=next_node;

END WHILE

Launch supervised learning (d, s);

Die

END

Launch machine learning (source: s, destination node: d)

WHILE (current node ≠ source node)

Choose next node by popping the stack

Update the cloud service attack dataset

Update the dataset table (T*sd*)

END WHILE

END

UPDATE THE dataset (T*sd*)

IF (T*sd*<=T_Good*sd*)

P*hd*1 ←

P*nd*0 ,ɣ n ≠ h, n ε Nk /* h is the node "come from" , k is the current node, Nkis the set of neighbors nodes and packet is the path or sub path destination */

ELSE

P*hd*1P*hd* + ←

Generate threat alert

Block the packet transmission

end

Performance Evaluation

Table 2: Cloud service attacks detection using machine learning

True Positive (TP)	2304
False Positive (FP)	52293
False negative (FN)	0
True negative (FN)	587033
Total No of cluster	641630

Accuracy = (2304 + 587033) / 641630 = 0.9185

The accuracy of the malware threats detection using machine learning is 92%

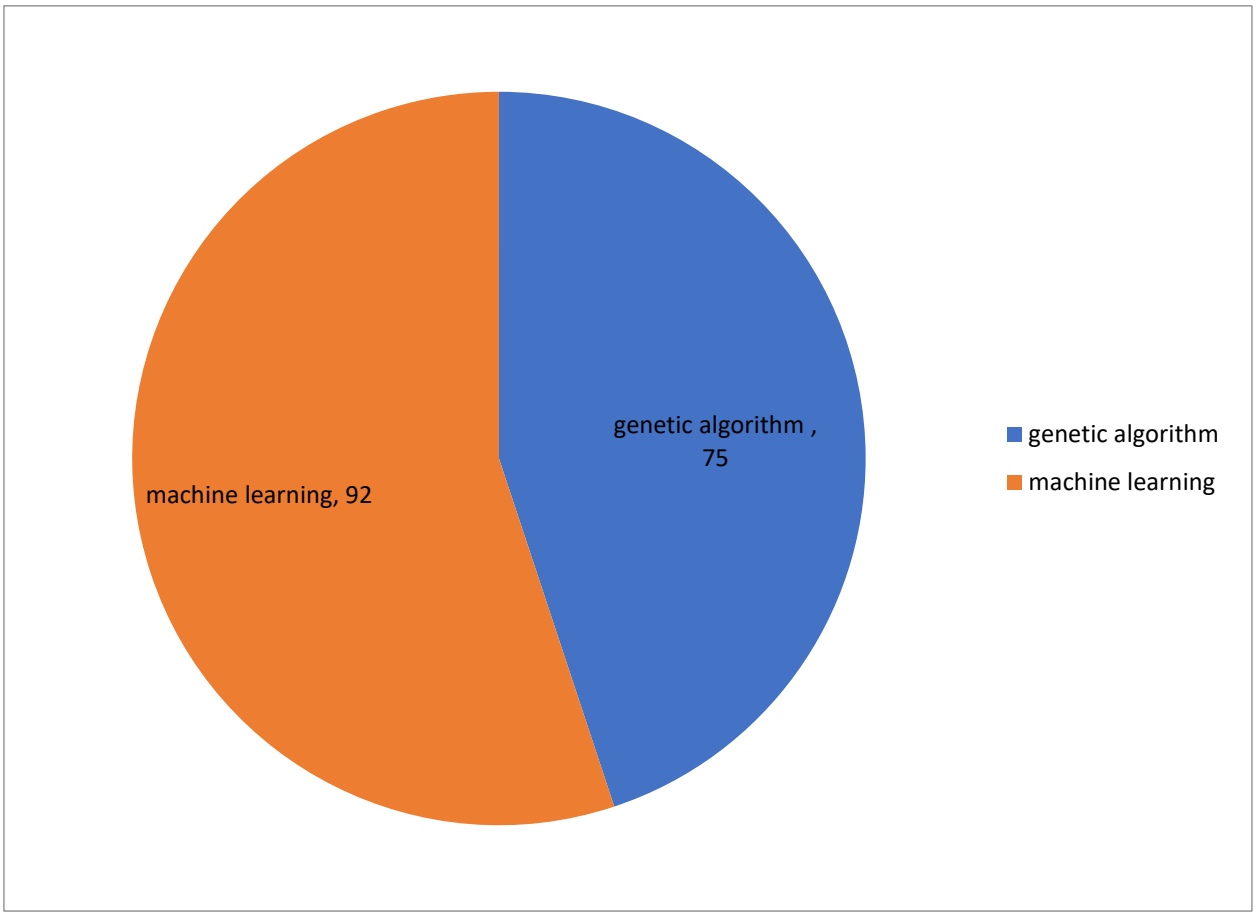

Figure 7: Performance evaluation

To evaluate our system, we focused in a major indicator of performance, which is the accuracy in detection rate and false positive rate. The evaluation of our system is done based on confusion matrix using measures 'Accuracy'. Using our approach, we have been able to achieve best results with detection rate and accuracy of 2% when we have used machine learning as shown in figure 4.12. This is better than 75% accuracy obtained with genetic algorithm.

Conclusion

The research conducted in this work initially provided an explanation pertaining to the key concepts of Cloud Computing and further provided a literature review with the aim of examining the cloud's security gaps. There are a number of different gaps in the security of Cloud Computing, which notably impacts all of its layers. Importantly, the decision to focus on network security was based on its practical importance as a fundamental aspect of the Cloud, meaning that any weaknesses present in the network have a profound and direct impact on the overall security of the Cloud. As has been highlighted throughout the literature, a number of different intrusions and attacks can impact overall network security, meaning Cloud network security is enhanced via the adoption of more commonplace defence approaches, including IDSs and firewalls, for example. With this noted, and in an effort to further reduce the scope of this work, the decision was made to utilise machine learning owing to the inability of firewalls to identify complex attacks, such as those of DoS and DDoS, and also insider attacks. In considering the aim of the research conducted in this research work focused on improving the security of the Cloud via the application of machine learning, a review pertaining to the use of supervised learning has been carried out in line with intrusion detection in order to build an intelligent system that has the ability to improve IDS performance. An intrusion detection problem, process anomaly detection, and a review of current research in this area were carried out. A cloud service attack detection and prevention system was built using machine learning algorithm. The system developed was carried out using php-mysql and java script. The software developed was tested and it was able to achieve 92% accuracy in cloud service attack detection and prevention in a network.

References

Dunlop, M.; Groat, S.; Shelly, D. (2020) GoldPhish: Using Images for Content-Based Phishing Analysis, Internet Monitoring and Protection (ICIMP), 2020 Fifth International Conference, pp.123-128, May 2020.

Jun, F.; Yu, C.; Wei-Shinn, K.; Pu, L. (2019) Analysis of Integrity Vulnerabilities and a Non-repudiation Protocol for Cloud Data Storage Platforms, Parallel Processing Workshops (ICPPW), 2019 39th International Conference, pp.251-258, Sept. 2019.

Engen, V. (2020) Machine Learning For Network Based Intrusion Detection, Int. J., 2020

Folasade, A. and Blaise, O. (2019) Brute-Force Attack Prevention in Cloud Computing Using One-Time Password and Cryptographic Hash Function. International Journal of Computer Science and Information Security (IJCSIS), Vol. 17, No. 2, February 2019

Jaspreet, K., and Rupinder, S. (2015). Prevention of DDoS and Brute Force Attacks on Web Log Files using Combination of Genetic Algorithm and Feed Forward Back Propagation Neural Network. International Journal of Computer Applications, 120(23), 0975-8887.

Carlisle, A., and Guy-Vincent, J. (2020). Lightweight protection against brute force login attacks on web applications. School of Information Technology and Engineering.

Modi, C., Dhiren, P., Bhavesh, B., Avi, P. and Muttukrishnan, R. (2013) A Survey on Security Issues and Solutions at Different Layers of Cloud Computing. The Journal of Supercomputing, 63, 561-592. https://doi.org/10.1007/s11227-012-0831-5

Mobin, J., and Vern, P. (2018). Detecting Stealthy, Distributed SSH Brute-Forcing. Association of Computing Machinery. doi:10.1145/2508859.2516719

Bahaa, Q. M. (2012). Preventing brute force attack through the analyzing log. Iraqi Journal of
Nitesh, B. (2017) Mitigating Distributed Denial of Service Attack in Cloud Computing Environment using Threshold based Technique. Indian Journal of Science and Technology, 3, 1-7.

Satomi, H., and Yuki, U. (2014). Detection of Novel-Type Brute Force Attacks Used Ephemeral Springboard IPs as Camouflage. Journal of Advances in Computer Networks, 2(4).