A Supervised Machine Learning Model for Detecting Network Attacks

Kenosi S. Mhlanga [1], Fungai Mukoko[2]
[1]*Computer Science Department, Harare Institute of Technology*
[2]*Computer Science Department, Harare Institute of Technology*

***Abstract*—This study developed a supervised machine learning model to enhance security in cloud computing. An in-depth examination of the present cloud computing security architecture was conducted to better understand cloud security issues. The model used labelled data to detect and classify traffic as either normal or anomaly. The model developed performed better than the traditional techniques of traffic attack detection based on historical data of previously detected attacks to explore and learn the patterns of attacks such that the model can predict and classify traffic as an attack or not. The study developed and implemented five supervised machine learning models: Logistic regression, Naïve Bayes, XGBoost Classifier, LightGBM model, and the Support vector machine classifier. The target feature was the classified traffic class which is either 'normal' or 'anomaly'. The two classes were encoded during feature engineering to produce two numerical traffic class codes: ' 0' for normal traffic and '1' for anomaly traffic. Coding was done using one-hot encoding. The results showed that the XGBoost Classifier model performs better than the other models as evaluated using four performance metrics namely, precision, f1 score, recall score and accuracy. The XGBoost scored the highest with a score of 100% on each metric and a reasonably low false positive of 15 entries. The study, therefore, concludes that the XGBoost classifier model is the best model to use in-network attack detection.**

***Index Terms*— Cloud computing, cloud security, supervised machine learning, classification algorithm, network, network attack.**

I.INTRODUCTION

The birth of cloud computing has significantly changed the way people apply technology in their day-to-day routines especially organisations as they seek increased computing power as well as portable services which could be acquired using traditional technologies in computing. However, migration to the cloud has also come with challenges to the users which includes security issues as it has presented new avenues of attacks that attackers exploit for their benefit. This study, therefore, focuses on the design and development of a machine learning-based cloud security system for the detection and classification of networks in cloud computing to improve security in cloud computing.

II.BACKGROUND TO THE STUDY

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet ("the cloud") to offer faster innovation, flexible resources, and economies of scale [1]. The emergence of Cloud Computing (CC) and the countless apps that support it has accelerated the evolution of information technology. CC allows users to disseminate resources over a network, allowing them to interact with any of them as needed and with flexible access. As a result, consumers' convenience and security are enhanced because they are no longer required to keep data on their own [2]. Companies can focus more on research and development of their products if they can buy into cloud services offered by specialized cloud service providers, rather than spending time and energy designing secure data storage facilities, purchasing hardware for that purpose, or training staff to follow procedures.

The various characteristics and benefits of cloud computing, such as improved efficiency, lower costs, expanded accessibility, reliability, and the flexibility to manage and scale systems, make it particularly appealing to a wide range of businesses and organizations across a wide range of industries [3]. However, the increased dependency on the cloud has created a series of security challenges which has

manifested through increased attacks on the platforms. Cloud security has thus become a major worry for cloud consumers. Because cloud computing relies significantly on user trust, there is concern that businesses may be exposed to new dangers and vulnerabilities. Furthermore, intruders or hackers could infiltrate cloud technology, giving them access to crucial data in the cloud that belongs to others [4]. An infiltration or attack on the cloud can have far-reaching consequences to the organization's being.

Because the majority of attackers are likely to target networks with the most users and automated and accessible services, as well as networks transporting the most data, they must be thoroughly evaluated, and all risks should be reduced whenever possible. As noted in [2], some of the most common and severe cybersecurity assaults on networks in various industries have been observed in the last two years. Network breaches and data security concerns are expected to become more common, according to security experts [5]. Every hour and day, various types of assaults are launched against computer networks, posing a substantial threat. They send out fresh attacks and trends, and these attacks go after every open port on the network in such high volumes that the traditional IDSs find difficult to process [6]. Several tools, such as network mapping and vulnerability scanning, were created for this purpose. Machine learning (ML) is a relatively new and widely used technology for feeding the Intrusion Detection System (IDS) with information about malicious network traffic [7]. Its strength in managing big data generated by network traffic makes it effective in detecting intrusions. However, the quality of the dataset used to train machine learning models is critical to the model's detection efficiency [8]. This study presents a supervised machine learning model for detecting malicious network traffic in CC to improve Cloud security. This detection technique makes use of a dataset made up of harmful and non-malicious traffic extracted from a Cloud Service Provider based in Zimbabwe.

III.PROBLEM STATEMENT

Despite the use of numerous strategies, most companies' servers, websites, and personal accounts continue to be targeted. Intrusion prevention systems are designed to detect and prevent intrusion attempts; however, they are unable to detect harmful traffic hidden within a packet that is considered legitimate. Signature-based Intrusion Detection Systems (IDS) can detect known assaults with ease, but they are unable to detect novel attacks for which no pattern exists. Anomaly Detection Systems (ADS) compare new behaviour to a known activity to detect unknown assaults. This method allows for the detection of previously unknown threats, but it also increases the chance of lawful behaviour being categorized as harmful. This shows a need for more research on the best effective strategy for dealing with these problems.

IV.LITERATURE REVIEW

The most fundamental aspect of the Cloud is its component-centric nature, which is recognized as providing several benefits, including customizability, extensibility, reusability, scalability, and substitutability, the latter of which includes alternative adoptions, runtime component replacements, and dedicated interfaces, all of which have been highlighted. In the study [9], the major focus of Cloud Computing was investigated, as well as the key differences that could be identified when comparing Grid Computing to the Cloud. Importantly, the notion of Cloud Computing in the field of computer science was defined in the work authored by [7], [8] through the presentation of many definitions.

Cloud Computing, on the other hand, was defined by [10] as an IT implementation architecture centred on virtualisation, in which diverse applications, data, and infrastructure-based resources are applied through the internet, 'as a distributed service by one or many different service providers. According to [9], such services are scalable on-demand and have a pricing structure based on a pay-per-use model. In the same vein, [11] claims that cloud computing makes use of virtualization technologies to achieve the goal of delivering computing resources as a valuable service.

When comparing Autonomic Computing, Grid Computing, and Cloud Computing, several elements are recognized as comparable; yet, there are some differences between the three. To define Cloud Computing, the US National Institute of Standards and Technology (NIST) has supplied a definition that

has gone through extensive development and has become a de-facto standard [12]. As a result, this definition, which is the most widely cited, highlights Cloud Computing as a framework that enables convenient, on-demand, universal access to a shared number of resources, such as applications, networks, servers, services, or storage, all of which is delivered and released in a time-effective manner without tying up resources.

Classification of Cloud attacks

The point has been made that a cloud attack can have a substantial influence on both service and network, with the attacker adopting media and network resources, resulting in a reduction in service performance, with the prospect of the network as a whole collapsing. Interaction, penetration, and mechanism are the three types of attacks that have been identified [13].

i)Interaction type

This type of attack is distinguished by the interaction between the attacker and the network environment, with such attacks being classified as either passive or aggressive [14]. In the case of the former, a large amount of critical data is obtained via tapping into traffic streams, such as by idle scan, port scanner, or wiretapping, for example. The attacker is known to influence the functioning of system resources or may otherwise choose to reconfigure them, such as through the use of ARP positioning, Denial of Service (DoS) attacks, Man-in-the-middle attacks, or Spoofing, in the case of the latter attack (active attacks). Importantly, such attacks are frequently difficult to detect because the attacker leaves very little evidence [15].

ii)Penetration Type

Penetration is found in both insider and outer attacks, with insiders being approved users who use their services to carry out unlawful or otherwise destructive activities or exploit other users' accounts [16]. In the case of an outsider attack, the attack is launched from outside the network's perimeter, using sensitive information gathered through scanning or probing attacks to launch genuine attacks later [17].

iii)Mechanism type

The attack can be classified into one of the following categories based on the numerous tactics and mechanisms used during initiation:

a) User to Root (U2R) attack

Attackers want Higher-level privilege to get system access and control by gaining login access and therefore bypassing the standard authentication process [18].

b) Remote to Local (R2L) attack

After successfully evading the regular authentication process, programs and commands with local machine privileges are executed on the victim host.

c) Probe or Scanning attacks; To get

access to network resources, such assaults explore networks seeking flaws or points of entry [19].

d) Denial of service (DoS) attacks

This attack has an effect on service availability by refusing or otherwise restricting users' access to a system's resources, such as bandwidth, buffers, memory, and/or processing capability. When attempting to make an attack successful, it is typical for software flaws to be used as the target, with changes to the way a system is set up and resources pushed to their limits [20]. ICMP Nukes, Land Attack, the Ping of Death, Teardrop, and modifying the settings of a hacked router are examples of such attacks.

e) Worm or Virus

Through the spread of malicious code across a host or network, this type of attack aims to cause data loss, theft, and malfunction [21].

Nonetheless, attacks can be classified as Cloud Computing surface attacks. As shown in Figure 1, a total of six possible Cloud surface assault categorizations have been offered in study of [21].

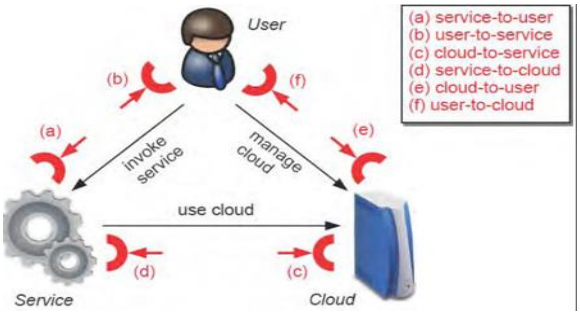

Figure 1: Cloud service attack categorization [7]

Furthermore, according to [18], attacks are classified as either host or network attacks depending on the attacker's behaviour in terms of the weakness exploited or the sort of technique used:

a. Host-based attacks

Attacks like these can happen as a result of flaws in apps or operating systems. In this regard, buffer overflow, format string, and rootkit are only a few instances [21].

b. Network-based attacks

Data modification, DoS attacks, eavesdropping, identity spoofing, IP address spoofing, and man-in-the-middle are some examples of network attacks in this case, with data modification, DoS attacks, eavesdropping, identity spoofing, IP address spoofing, and man-in-the-middle being some examples of network attacks [1].

V. METHODOLOGY

Five models were tested to detect network attacks and their metrics compared to check the best performing model. The models were fitted with the same one-hot encoded and scaled data. Tested models are the Support Vector Classifier (SVC), Logistic Regression (LR), Bernoulli Naïve Bayes (BNB), XGBoost Classifier (XGBC) and Light GBM Classifier (LGBMC).

a. Dataset

The dataset used contains network traffic collected by one of the Network service providers. The data is divided into Training instances of 25,192 with 42 features and Testing instances of 22,544 with 41 features (less the Class). The label is the class with either normal or anomaly, which is used for classification. A sample of dataset is shown in Table 1.

Table 1: Dataset Sample

dst_host_same_src_port_rate	dst_host_srv_diff_host_rate	dst_host_serror_rate	dst_host_srv_serror_rate	dst_host_rerror_rate	dst_host_srv_rerror_rate	class
0.17	0.00	0.00	0.00	0.05	0.00	normal
0.88	0.00	0.00	0.00	0.00	0.00	normal
0.00	0.00	1.00	1.00	0.00	0.00	anomaly
0.03	0.04	0.03	0.01	0.00	0.01	normal
0.00	0.00	0.00	0.00	0.00	0.00	normal

There is a slight imbalance in the target column "class" of the dataset. Most of the ICMP traffic had anomalies and most of the UDP traffic was normal, while the distribution was almost equal in the case of TCP. The traffic distribution based on flags was also uneven where most of it had SF (Sign Flag). Most of the traffic with SF was normal, while that had S0 flag had an anomaly. Most of the traffic recorded was unique. The target distribution is shown in Figure 2.

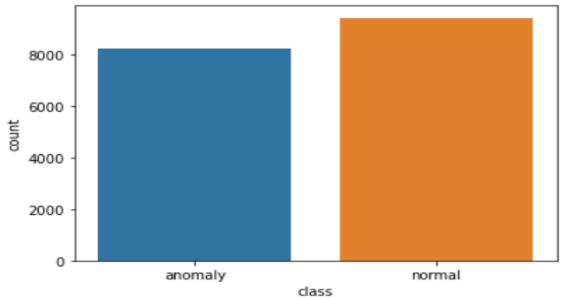

Figure 2: Target distribution

VI. MACHINE LEARNING ALGORITHMS USED

The study adopted an ensampling approach in which five algorithms were used to classify the attacks with the best algorithm being retained for implementation. The comparison is on support vector classifier, logistic regression, Naïve Bayes Classifier, XGBoost Classifier and the LightGBM Classifier. The models are shown in Figure 3.

```
In [37]:
# Models

# SVC Model
svc = SVC(random_state=SEED)

# LogisticRegression Model
lr = LogisticRegression()

# Gaussian Naive Bayes Model
bnb = BernoulliNB()

# Train XGBoost Classifier
xgbc = XGBClassifier(eval_metric="logloss", random_state=SEED)

# Train LightGBM Classifier
lgbmc = LGBMClassifier(random_state=SEED)
```

- Testing on 5 models to find the best model.

Figure 3: Models used for comparison

The models were trained, and the validation was done using the same data as shown in Figure 4.

```
In [38]:
#Model Testing on Validation Data
models = {}
models['SVC']= svc
models['LogisticRegression']= lr
models['Naive Bayes Classifier']= bnb
models['XGBoost Classifier']= xgbc
models['LightGBM Classifier']= lgbmc
scores = {}
for name in models:
  scores[name]={}
  for scorer in ['precision','recall']:
    scores[name][scorer] = cross_val_score(models[name], X_train_scaled, y_train, cv=10, scoring=scorer)
```

Figure 4: Model Validation

After validation, the models were subjected to prediction using test data to find the best performing model for the use case. The best model will be able to classify network attacks as normal or anomaly from the given dataset. Figure 5 shows the models prediction.

```
models = {}
models['SVC']= svc
models['LogisticRegression']= lr
models['Naive Bayes Classifier']= bnb
models['XGBoost Classifier']= xgbc
models['LightGBM Classifier']= lgbmc
preds={}
for name in models:
    models[name].fit(X_train_scaled, y_train)
    preds[name] = models[name].predict(X_test_scaled)
print("Predictions complete.")
```

```
Predictions complete.
```

Figure 5: Model predictions

VII.RESULTS OF ALGORITHM PERFORMANCE

A confusion matrix for each model was obtained after exposing the model to test data for prediction as shown in Figure 6.

```
def line(name,sym="*"):
    return sym*(25-len(name)//2)
target_names=["normal","anamoly"]
for name in models:
    print(line(name), name, 'Model Testing', line(name))
    print(confusion_matrix(y_test, preds[name]))
    print(line(name,'-'))
    print(classification_report(y_test, preds[name], target_names=target_names))
```

Figure 6: Confusion Matrix for each model

The performance of the five algorithms is shown in Figures 7-11 which are snippets from the python script results. Figure 7 shows the performance metrics of the SVC model.

```
************************ SVC Model Testing **********
[[3981   61]
 [  27 3489]]
------------------------
              precision    recall  f1-score   support

      normal       0.99      0.98      0.99      4042
     anomaly       0.98      0.99      0.99      3516

    accuracy                           0.99      7558
   macro avg       0.99      0.99      0.99      7558
weighted avg       0.99      0.99      0.99      7558
```

Figure 7: SVC Model Testing Results

The results of the SVC model show that the model classified the traffic with an accuracy of 99%, precision on normal traffic of 99% and 98% on anomaly traffic. The model had recall and f1-scores of 98% and 99% respectively. The confusion matrix of the model shows that it had an error in making predictions by making 27 false positives and 61 false negatives. This shows that the model failed to correctly classify 88 network traffic recorded in the dataset.

Figure 8 shows the performance metrics of the Logistic regression model.

```
*************** LogisticRegression Model Testing ***
[[3852  190]
 [ 246 3270]]
----------------
              precision    recall  f1-score   support

      normal       0.94      0.95      0.95      4042
     anomaly       0.95      0.93      0.94      3516

    accuracy                           0.94      7558
   macro avg       0.94      0.94      0.94      7558
weighted avg       0.94      0.94      0.94      7558
```

Figure 8: Logistic regression Model Testing Results

The results show that the Logistic regression model classified the traffic with a precision score of 94% on normal traffic and a 95% on anomaly traffic, recall score of 95% on normal traffic and 93% on anomaly traffic, F1-score of 95% on normal traffic and 94% on anomaly traffic as well as the accuracy of 94% on both classes. The confusion matrix of the model also shows that the model made 246 false positives and 190 false negatives giving a total of 428 false predictions. The number of false predictions is too high that it affects the effectiveness of the model. The false predictions of the Logistic regression model are higher than the false predictions of the SVC model indicating a better performance of the latter concerning the metrics of the confusion matrix.

Figure 9 shows the performance metrics of the Naïve Bayes Classifier model.

```
************** Naive Bayes Classifier Model Testing **
[[3824  218]
 [ 525 2991]]
--------------
              precision    recall  f1-score   support

      normal       0.88      0.95      0.91      4042
     anomaly       0.93      0.85      0.89      3516

    accuracy                           0.90      7558
   macro avg       0.91      0.90      0.90      7558
weighted avg       0.90      0.90      0.90      7558
```

Figure 9: Naïve Bayes Classifier Model Testing Results

Metrics of the Naïve Bayes model show that the model classified the traffic with an accuracy of 90% on both classes, precision of 88% on normal traffic and 93% on anomaly traffic, recall score of 94% on normal traffic and 85% on anomaly traffic. The confusion matrix shows that the model was classified with 525 false positives and 218 false negatives making a total of 743 false classifications. The false classifications of the Naïve Bayes model are higher than the false classifications of both the SVC and Logistic regression models indicating that it performs less as compared to the two.

Figure 10 shows the performance metric results of the XGBoost Classifier model.

```
********** XGBoost Classifier Model Testing ****
[[4037    5]
 [  10 3506]]
----------
              precision    recall  f1-score   support

      normal       1.00      1.00      1.00      4042
     anomaly       1.00      1.00      1.00      3516

    accuracy                           1.00      7558
   macro avg       1.00      1.00      1.00      7558
weighted avg       1.00      1.00      1.00      7558
```

Figure 10: XGBoost Classifier Model Testing Results

The performance metrics show that the model is classified with maximum scores on all the metrics used as recorded on both the normal and anomaly traffic. The confusion matrix of the model shows that the model made 10 false positives and 5 false negatives making a total of 15 false classifications. In this regard, the XGBoost classifier performed better than the Naïve Bayes, Logistic Regression and SVC models. Figure 11 depicts the outcome of LightGBM classifier metrics.

```
********** LightGBM Classifier Model Testing ***
[[4032   10]
 [   9 3507]]
----------
              precision    recall  f1-score   support

      normal       1.00      1.00      1.00      4042
     anomaly       1.00      1.00      1.00      3516

    accuracy                           1.00      7558
   macro avg       1.00      1.00      1.00      7558
weighted avg       1.00      1.00      1.00      7558
```

Figure 11: LightGBM Classifier Model Testing Results

The metrics show that the model predicted the traffic classes with a maximum score of 1 which is the highest possible score on all the metrics used. The metrics are similar to the scores of the XGBoost model indicating that the two models equally outperformed the other models used. However, the confusion matrix of the LightGBM Model indicates that the model made 10 false positives and 9 false negatives making a total of 19 false predictions. Although the total number of false predictions of the LightGBM model is lower enough to be satisfactory, the number of false predictions is higher than the false predictions recorded in the XGBoost model. This shows that the XGBoost outperformed the LightGBM model and the other models at large.

VIII.ANALYSIS OF RESULTS

The results of the performance of the models have been captured and visualized in bar graphs in Figures 7-8. Figure 7 show the results of the F1 scores of the models used.

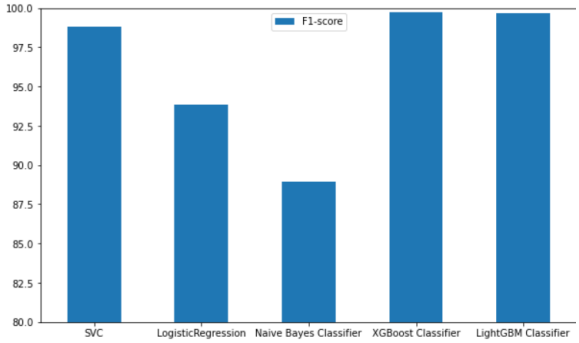

Figure 12: Comparison of F1 scores of the five algorithms

The results show that the XGBoost classifier model had the highest F1 score followed by the LightGBM model, followed by the SVC classifier and next is the Logistic regression model while the Naïve Bayes classifier is the least in performance using the F1 score. Another comparison of the model performances is depicted in Figure 8 which compares the model performances using precision and recall as the comparative metrics.

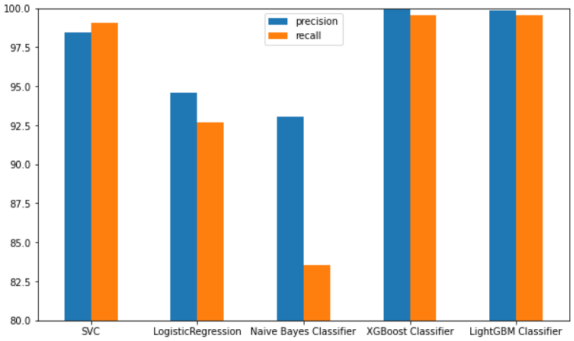

Figure 13: Precision and recall of the five algorithms

The results show that the XGBoost classifier outperforms all the other models in both the precision and recall scores followed by the LightGBM model, followed by the SVC model and next is the Logistic regression and lastly the Naïve Bayes model.

IX.CONCLUSION

The study concludes that the XGBoost Classifier is the best performing model in predicting network attacks using labelled data. The study also concludes that, in general, gradient boosting models perform better in detecting and classifying network traffic

concerning attacks as compared to other classification models.

ACKNOWLEDGEMENT

My acknowledgements go to my Supervisor Mr. F.D Mukoko for spending his precious time to guide me through the process. A special acknowledgement also goes to my family for their moral support. May God bless you all.

REFERENCES

[1] Alam, "Cloud Computing and its role in the Information Technology," IAIC Transactions on Sustainable Digital Innovation (ITSDI), vol. 3, no. 2, pp. 108-115., 2021.

[2] S. Mthunzi, "Cloud computing security taxonomy: From an atomistic to a holistic view," Future Generation Computer Systems, vol. 107, pp. 620-644., 2020.

[3] E. Margherita, "IS in the cloud and organizational benefits: an exploratory study," in In Exploring Digital Ecosystems, Cham, Springer, 2020, pp. (pp. 417-428).

[4] S. Namasudra, "Data access control in the cloud computing environment for bioinformatics," International Journal of Applied Research in Bioinformatics (IJARB), vol. 11, no. 1, pp. 40-50, 2021.

[5] P. Kumar, "Exploring data security issues and solutions in cloud computing," Procedia Computer Science, vol. 125, pp. 691-697., 2018.

[6] H. Tabrizchi, "A survey on security challenges in cloud computing: issues, threats, and solutions," The journal of supercomputing, vol. 76, no. 12, pp. 9493-9532., 2020.

[7] M. Mehmood, "A review of machine learning algorithms for cloud computing security," Electronics, vol. 9, no. 9, p. 1379, 2020.

[8] M. Rabbani, "A hybrid machine learning approach for malicious behaviour detection and recognition in cloud computing," Journal of Network and Computer Applications, vol. 151, no. 3, p. 102507, 2020.

[9] M. Ding, "When machine learning meets privacy: A survey and outlook," ACM Computing Surveys (CSUR), vol. 54, no. 2, pp. 1-36, 2021.

[10] T. Alam, "Cloud Computing and its role in the Information Technology," IAIC Transactions on Sustainable Digital Innovation, vol. 1, no. 2, pp. 108-115, 2021.

[11] N. &. J. A. Subramanian, "Recent security challenges in cloud computing," Computers & Electrical Engineering, vol. 71, no. 56, pp. 28-42, 2018.

[12] A. Piter, "Exploring Cloud-Based Platforms for Rapid Insar Time Series Analysis," The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 43, pp. 171-176, 2021.

[13] G. Shidaganti, "Scef: A model for prevention of ddos attacks from the cloud.," International Journal of Cloud Applications and Computing (IJCAC), vol. 10, no. 3, pp. 67-80, 2020.

[14] F. Hamdani, "Detection of DDOS attacks in cloud computing environment. In 2019 International Conference on Intelligent Computing and Control Systems (ICCS)," IEEE, pp. (pp. 83-87), 2019.

[15] N. Agrawal, "Defense mechanisms against DDoS attacks in a cloud computing environment: State-of-the-art and research challenges," IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3769-3795, 2019.

[16] L. Alhenaki, "A survey on the security of cloud computing. In 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)," IEEE, pp. (pp. 1-7), 2019.

[17] S. Tian, "Fingerprinting cloud FPGA infrastructures," In Proceedings of the 2020 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. (pp. 58-64), 2020.

[18] K. Virupakshar, "Distributed denial of service (DDoS) attacks detection system for OpenStack-based private cloud.," Procedia Computer Science, vol. 167, pp. 2297-2307, 2020.

[19] G. Zhang, "Detection of hidden data attacks combined fog computing and trust evaluation method in sensor-cloud system," Concurrency and computation: practice and experience, vol. 33, no. 7, pp. 1-1, 2021.

[20] M. Zareapoor, "Advance DDOS detection and mitigation technique for securing cloud," International Journal of Computational Science

and Engineering, vol. 16, no. 3, pp. 303-310, 2018.

[21] M. Idhammad, "Detection system of HTTP DDoS attacks in a cloud environment based on information theoretic entropy and random forest," Security and Communication Networks, 2018.