Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Detecting API Abuse and Authentication Attacks Using Artificial Intelligence in Cloud Applications

Sai Kiran Arcot Ramesh
Campbellsville University
https://orcid.org/0009-0003-0633-4618

ARTICLE INFO

Received: 06 Nov 2023

Revised: 20 Dec 2023

Accepted: 28 Dec 2023

ABSTRACT

With cloud applications playing an important role in current businesses, protecting APIs and authentication systems has emerged as the most crucial aspect in preventing abuse and cyberattacks. The paper introduces an innovative hybrid AI-based API abuse and authentication attacks detector in cloud computing. The recommended model is based on the advantages of the random forest (RF) and the Long Short-Term Memory (LSTM) network to identify both the static and the sequential attack patterns. The Random Forest model is very efficient in detecting high-volume anomalies, including rate-limiting attacks and DoS attacks, whereas the LSTM network is very efficient at detecting the time dependencies in the data, being especially well-suited to detect brute-force attacks and credential stuffing attacks. The hybrid model proved to be a better model than the single models with 96.5% accuracy, 93% precision and 98% recall and AUC, 0.97. The model was shown to be superior to the traditional rule-based systems, pointing to the benefits of ensemble learning and deep learning to address the complexity and continually changing nature of current cyber threats. The model has challenges observed in terms of computational complexity as well as real-world datasets, despite performance which is promising. The next step in work will be to increase scalability, solve false positive and false negative rates, and add other types of attacks to a more complete security solution. This study contributes to a powerful, AI-based method of real-time security of cloud applications, and it would offer a strong opportunity to enhance the detection and prevention of API abuse and authentication attacks.

Keywords: API abuse, Authentication attacks, Cloud security, Machine learning, Random Forest, LSTM, Hybrid model, Brute-force attacks, Credential stuffing, Anomaly detection

1 INTRODUCTION

Cloud applications have emerged as part of the business in recent years, providing scalable and flexible applications for most of the services. As cloud systems have expanded at a very quick rate, there has been a significant increase in reliance on cloud applications and APIs, with a corresponding rise in the security concerns surrounding these environments [1]. Particularly, APIs are essential since they are the entry points to interactions between services and applications. Nevertheless, this large dependency on APIs has also predisposed APIs to cyberattacks. In this respect, strong security frameworks have become the key to protecting important data and preserving the integrity of the cloud environment. The safety of APIs, along with the authentication procedures that they are based on, should be strengthened to eliminate any forms of misuse and unauthorised access, which may result in major data breaches and losses [2].

API abuse and authentication attacks are some of the most urgent issues in cloud security that need to be prevented. The concept of API abuse is the abuse or overuse of APIs, which are mainly performed in a manner that causes overloading of the system or huge amounts of data to be retrieved illegally [3]. These may be scraping (unauthorised extraction of data), rate-limiting bypass, and denial of service (DoS) attacks in which the service is overwhelmed with a high number of requests. An infamous case of API misuse was with the API of Facebook, where attackers used the API to scrape personal information of users without their consent, which resulted in the famous Cambridge Analytica scandal [4]. Authentication attacks, however, involve the login systems of the applications, on which a vulnerability in verifying the identity of a user is exploited. Credential stuffing is one of the most frequently used attack tactics in

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

which the attackers submit logins and passwords that have been previously leaked into accounts in order to gain unauthorised access. The other attack is that of brute-force attacks, in which an attacker tries to deduce passwords by repeatedly guessing them until they find the right one [5]. Such attacks have been largely prevalent in banking applications, as bots are constantly trying to crack into an account by trying many credentials. Such attacks may lead to huge losses to businesses in terms of financial losses and reputation.

With the emergence of artificial intelligence (AI) and machine learning (ML) technologies, there are new opportunities to fight such attacks. The use of AI-based solutions, specifically those based on machine learning classifiers, has demonstrated that AI applications can significantly improve the detection and response of security systems [6]. Machine learning techniques, including the Random Forest, Support Vector Machines (SVMs), and Neural Networks, have proven their efficacy in the discovery of trends of malicious behaviour, even in large and intricate systems. These models can be trained to distinguish between the normal and abnormal API behaviour, which enables the detection of abuse attempts early and the response in real-time [7]. In particular, the use of Random Forest is most frequently explained by the fact that it is able to work with large data sets and its strong efficiency in the classification process. The neural networks and more specifically the deep learning models can learn complex patterns in sequential data, which is why they are optimal at detecting authentication attacks, since the sequence of attempts to access the system and user behaviour should be analysed over time [8].

Although the API and authentication security have been advanced, the existing detection systems have a number of limitations. Old signature-based detection systems that use fixed patterns of known attacks are generally not suitable in the context of dealing with new attacks or new vulnerabilities [9]. Such systems are only able to know attacks that they have already seen, and thus, the emerging or new attacks remain undetected. As an example, zero-day exploits, which are vulnerabilities that are not known, are not visible to signature-based systems, since they do not have attack signature definitions. Moreover, such systems are not always able to identify the nuanced abnormalities in user behaviour, including credential stuffing or low-volume brute-force attacks, which can be mistakenly regarded as part of a regular usage pattern, but are signs of malicious intent [10]. As a result, the modern detection approaches have an evident gap, which is sought to be filled through this study, which examines more sophisticated AI-based models that can detect known and unknown threats in real-time.

The main goal of this research was to suggest a new hybrid AI system to identify API abuse and authentication attacks in the cloud. The model combines both supervised and unsupervised learning methods to increase the accuracy of detection and be flexible to tackle different attack cases. The proposed model is expected to raise the detection capabilities of the traditional pattern of attacks, as well as new or never seen before anomalies in the use of API and authentication attempts, by integrating the power of multiple machine learning algorithms. The hybrid method uses the fact that Random Forest and Neural Networks can process large and high-dimensional data, but includes LSTM (Long Short-Memory) networks to learn sequential relationships in authentication logs. This is aimed at developing a system that is capable of reacting to malicious activities with high precision and low false positives, and this is a significant development over the traditional systems that merely use rule-based or signature-based approaches.

The study aims to fill the gap between the conventional cybersecurity techniques and the contemporary machine learning-based approaches, and provide a more dynamic and adaptive approach to cloud security. Such a system development is vital in the resilience of cloud applications to upcoming threats and in providing businesses the ability to offer a robust security solution capable of remaining scalable with increased utilisation of APIs and cloud-based services.

2 LITERATURE REVIEW

2.1 API Abuse Detection

In recent years, the discovery of API abuse has become a topic of great attention because APIs are becoming the focus of most cloud applications. Some of the most common and traditional methods of API abuse detection are rate-limiting and signature-based detection [11]. Rate-limiting, which is a form of limiting the number of requests an API may service in a particular time frame, can be used to prevent some forms of denial-of-service (DoS) attacks as well as overuse. On the contrary, signature-based techniques are based on the recognition of well-known attack patterns by comparing request logs with pre-established signatures [12]. Such approaches have been popular but have serious

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

shortcomings, especially in identifying new attacks or new attack tactics. Systems based on signatures are limited to detecting attacks that were already spotted, and hence they cannot be used in detecting new and novel attacks. As an example, scraping attacks, i.e., the unauthorised gathering of big datasets by an API, may go undetected unless the given patterns are familiar [13].

Contrarily, more flexible alternative methods, including k-nearest neighbours (k-NN) and k-means clustering, have been suggested as methods of anomaly detection. They are applied to identify unusual API usage even in cases where the attack patterns are not known, and therefore they can be used to identify abnormal practises of an API even in a large-dimensional feature space [14]. k-NN is one such technique, where the API requests are classified by similarity of features to those of similar previously seen requests. Yet there are also disadvantages to such methods, such as their inability to scale to large datasets and the existence of high false-positive rates in conditions that have high variance of normal traffic. Also, certain anomaly detection algorithms do not model the intricate connections among the various API calls or patterns that change over time, which may restrict their utility in practice [15].

2.2 Authentication Attacks

Credential stuffing and brute-force attacks are examples of authentication attacks that remain common in most applications, particularly those that deal with sensitive information or money. Credential stuffing happens when attackers utilise already acquired usernames and passwords of accounts to access them illegally [16]. This is an effective way of attack because many users use the same credentials on different platforms, and this attack method capitalises on this fact. On the other hand, forced attacks are those in which attempts to crack a password are carried out systematically by systematically increasing the possible combinations of characters [17]. Such attacks are computationally expensive but may be automated to be highly successful with time, particularly where poor passwords are used.

The conventional ways of identifying these attacks are based on rate-limiting and blocking of IP addresses. Such systems warn the accounts when they have already made several failed login attempts, based on the assumption that a series of wrongful attempts within a short period implies malicious intent [18]. Although this approach will stop a significant number of brute-force attacks, the approach can easily be defeated by attackers who employ methods, such as distributed botnets, whereby multiple IP addresses are utilised to carry out the attack. In addition, these conventional approaches are characterised by high false-negative rates, especially for low-frequency or long-duration attacks [19]. In a bid to enhance detection, scientists have developed AI-based methods, including logistic regression and decision trees. These approaches represent the dependence of various characteristics (e.g., time of login attempt, geographical location, user behaviour) and utilise historical data to identify the attempts to log in as either normal or malicious [20]. Although these models have a more data-driven approach, they also have a problem with adapting to changing attack strategies and large-scale datasets.

2.3 Machine Learning in Security

Machine learning (ML) has been a subject of cybersecurity implementation in the last ten years, with the approaches of supervised learning, unsupervised learning, and deep learning currently forming the main pivots in the process of API abuse and authentication attacks [21]. The use of supervised learning which models are trained on labelled datasets containing both normal and malicious data, has been largely utilised in the detection of security threats. Some of the common algorithms in this category are decision trees, random forests and support vector machines (SVMs). These models can train to differentiate between normal and malicious acts on the basis of historical data; however, they need labelled data sets and do not always cope with novel or changing forms of attacks [22].

Conversely, unsupervised learning approaches, like k-means clustering and autoencoders, are meant to identify anomalies without needing any labelled data [23]. They are especially helpful when it comes to detecting an attack pattern previously unknown, since they are concerned with learning the normal behaviour of users/APIs, and an unusual pattern of behaviour will be treated as a potential attack. As an example, autoencoders, which are a form of neural network that are applicable in anomaly detection, can be trained to compress and rebuild the input information [24]. In the event that the error in reconstruction is large, the input is a suspect and could be an attack. Although an unsupervised learning model has the benefit of detecting new threats, they have a higher rate of false positives because it may use the assumption that any deviation from their normal behaviour is malicious.

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

The latest developments in cybersecurity include deep learning methods, specifically, recurrent neural networks (RNNs) and long short-term memory (LSTM) networks [25]. They are effective at sequencing data and time; hence, these models are ideal to analyse user login sequences and API request sequences across time. LSTMs can learn long-range dependencies and will be useful in identifying brute-force attacks, which may comprise several failed logins separated by time [26]. Nevertheless, the models of deep learning are data-intensive and need considerable computational resources, which might constitute a drawback to most organisations, particularly when it comes to real-time detection.

2.4 Hybrid Models

To address the limitations of a single machine learning method, recent research has considered the application of hybrid models that integrate various machine learning methods to ensure greater scalability and robustness. In this direction, ensemble methods like stacking and boosting have become popular. In stacking, multiple base models are used to form predictions to make a final prediction (e.g. decision trees, random forests, and neural networks) [27]. The concept is that through the integration of several models, the strengths of all models can be availed, leading to improved generalisation and low overfitting. As an example, a Random Forest could be utilised to detect suspicious API traffic, and an LSTM model could be used to detect time patterns in the user login process [28]. The hybrid system can bring a more precise and all-encompassing method of detecting API abuse and authentication attacks by integrating these models.

AdaBoost and XGBoost Boosting techniques have also been used to solve security issues, especially when the data is skewed or the cost of a false alarm is great. The idea of boosting is to up-sample the instances in the dataset that are difficult to classify, and gradually, the model improves [29]. Boosting may be applied to API abuse and authentication attack detection to improve the detection rate of the rare but important attack cases and minimise the false negatives. Stacked generalisation or stacking has demonstrated favourable outcomes in integrating models that are strong in various dimensions of the detection task to produce a stronger system with the ability to perform well in complex real-world situations [30].

2.5 Research Gap

In summary, although different machine learning and artificial intelligence models have been suggested to detect API abuse and authentication attacks, there is still a definite gap in the research to tackle new attacks and intricate attack patterns, which are not adequately identified by traditional solutions. The signature-based detection systems still fail to work effectively against zero-day exploits and evolving attack strategies, and so adaptive machine learning models become very critical. Hybrids of two or more machine learning strategies show enormous potential for enhancing detection rates as well as lowering the number of false positives. It is suggested in this paper that a new hybrid AI model, which incorporates a combination of supervised learning, unsupervised learning, and deep learning, will enhance real-time detection of both API abuse and authentication attacks. The proposed model will attempt to fill the gaps in the traditional signature-based systems and the necessity of dynamic, adaptive, and robust security solutions in the contemporary cloud environment by exploiting the strengths of both methods.

3 METHODOLOGY

3.1 Dataset Description

For this research, two important datasets were used: API traffic logs and authentication logs. The study used the **API security: Access behavior anomaly dataset** of Kaggle to model API request behaviour. This data includes API traffic logs, including such important characteristics as timestamps, response time, request method, and status. These are important features that can be used to detect unusual usage patterns of the API, e.g., too many requests or violation of rate limits. Each data point in this dataset is associated with an API call, and features that characterise the request, and the response to the call can be used to identify both benign and malicious behavioural patterns. **Web Server Access Logs** were utilised in the detection of anomalies related to authentication. Such logs have information regarding the authentication efforts, both successful and unsuccessful attempts to the company, as well as attributes such as IP addresses, geographical locations, and the period during which the pupil has their account. Such characteristics can be used to identify brute-force and credential stuffing attacks. The logs present a wealth of

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

information to be used in training as well as testing machine learning models that can be used to identify abnormal behavioural tendencies in users as they use them over time.

3.2 Mathematical Model for Anomaly Detection

The aspect of the methodology is that the core is the feature space. X representing an API request or authentication attempt. Mathematically, let:

$$X = \{x_1, x_2, \dots, x_n\} \quad (1)$$

where $x_i \in \mathbb{R}^d$denotes feature vectors of the i-th request, and dis is the number of features, which in the case of API logs are request method, IP address, response time, and status code of an API call, and in the case of authentication logs, time, IP address, and successful or failed attempt. These are the features that are necessary to model normal behaviour and identify the abnormalities that may be either a sign of abuse or attack.

3.3 AI Models Used

3.3.1 Random Forest (RF)

Random Forest classifier, one of the most popular ensembles learning algorithms, was selected because it allows aggregating decision trees, so the risk of overfitting is reduced. Through separate decision trees in the forest, all decisions make predictions, and the overall decision is based on the majority vote of all decision trees.

The mathematical model of the prediction $\hat{y}$ of a specific test sample x is:

$$\hat{y} = \frac{1}{K}\sum_{k=1}^{K} h_k(x) \quad (2)$$

where $h_k(x)$represents the output of the k-th tree, and K denotes the total number of trees in the forest. The average predictions that are achieved by multiple decision trees in the random Forest algorithm enhance accuracy and robustness, particularly when there are noise or imbalanced data.

3.3.2 Long Short-Term Memory (LSTM) Networks

LSTM networks are a form of recurrent neural network (RNN) that was utilised to represent the sequential relationships in API request logs and authentication attempts. They are especially useful in identifying a time-based pattern, like brute attacks, when a sequence of failed logins in time might be a sign of bad actions.

The hidden state h_t at the time step t in an LSTM is updated using:

$$h_t = \sigma(W_h h_{t-1} + W_x x_t + b) \quad (3)$$

where $W_h$is the weight matrix for the previous hidden state, $W_x$is the weight matrix for the input at time t, $x_t$is the input at time t, and $b$is the bias vector. The model captures **long-term dependencies** and enables more accurate prediction of attack patterns in sequential data.

The loss function for training the LSTM is again the **binary cross-entropy loss**:

$$\mathcal{L}_{LSTM} = -\sum_{t=1}^{T}[y_t \log(\hat{y}_t) + (1 - y_t)\log(1 - \hat{y}_t)] \quad (4)$$

where $T$is the length of the sequence and $y_t$is the true label at time step t. This loss function helps the LSTM network to minimise the error in predicting the class labels for each time step in the sequence.

3.3.3 Hybrid Model

To enhance performance and strength, a hybrid model was developed based on the stacked ensemble of Random Forest and Neural Networks or LSTMs. The conceptualisation of this is to use the strengths inherent to both models: Random Forest to use high-dimensional and noisy features, and LSTM to learn sequential dependencies.

The last prediction $\hat{y}$ is calculated by summing up the predictions of several models as follows:

$$\hat{y} = \frac{1}{K}\sum_{k=1}^{K} f_k(x) \quad (5)$$

where $f_k(x)$ is the prediction of the k-th model (e.g., Random Forest, LSTM, Neural Network) and K Is the number

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

of models in the ensemble. It is hoped that such a combination will enhance accuracy and minimise false positives since the ensemble method will integrate the predictive ability of multiple classifiers to enhance resiliency to a range of attacks.

3.4 Feature Engineering and Preprocessing

The features that train the models are important with regard to the quality of the features. The feature engineering was conducted to retrieve the appropriate features of the raw API logs and authentication data. These were critical attributes such as request rate, response time, and pattern of IP address requesting API, time interval of login attempts and pattern of IP address requesting authentication logs.

3.4.1 Data Normalisation

Data normalisation of the input features was used to ensure that the models maximise their output, particularly numerical data such as response times and request rates. These features were normalised with the help of min-max scaling:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (6)$$

This guarantees that features are on a comparable scale, such that features with bigger scopes do not dominate the learning process by the model.

3.4.2 Handling Imbalanced Data

To overcome the problem of imbalanced classes (with anomalous requests being significantly less than normal ones), the study used the SMOTE (Synthetic Minority Over-sampling Technique) algorithm. SMOTE operates by generating fake samples of the minority group (i.e. attack patterns) by interpolating between existing samples. This assists in the balancing of the dataset and enhances the model performance through the increased examples of the model of the rare attack patterns.

3.5 Model Evaluation

The performance of the models was evaluated using several common metrics:

- **Accuracy**: Measures the overall percentage of correct predictions.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (7)$$

- **Precision**: The proportion of true positive predictions among all predicted positives.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (8)$$

- **Recall**: The proportion of true positive predictions among all actual positives.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (9)$$

- **F1-Score**: The harmonic mean of Precision and Recall, providing a balanced measure.

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

- **ROC-AUC**: Measures the model's ability to discriminate between normal and anomalous behaviours across different thresholds.

In order to test the strength and applicability of the models, k-fold cross-validation was employed, which is used to ensure that the models would perform in a certain manner with respect to various subsets of the data.

3.6 Tools and Environment

The Python language was used to implement the models, with the help of Scikit-learn, Keras, TensorFlow, and Matplotlib libraries to train the model, evaluate it, and visualise the results. The experiments were implemented using Google Colab that gives access to the powerful cloud resources to execute deep learning models at scale.

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

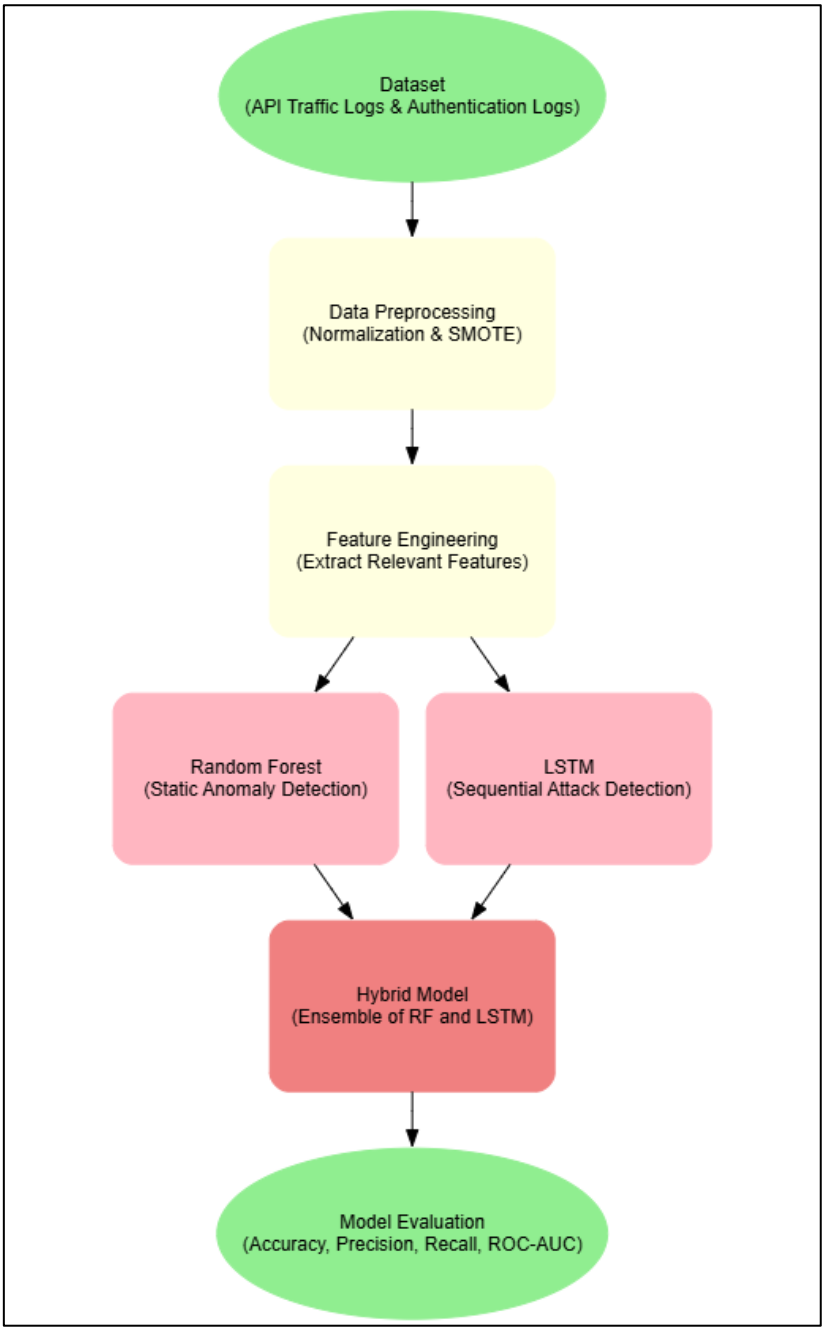

Figure 1. Proposed Methodology Pipeline

Figure 1 shows a stepwise procedure for identifying API abuse and authentication attacks. It starts with the preparation of data sets, during which API logs of traffic and authentication logs are gathered. The data is then pre-treated, and this includes normalisation and SMOTE to balance the data. It is followed by feature engineering to obtain important features to be analysed. Random Forests model is used to identify the presence of static anomalies, and LSTM is used to deal with sequential patterns of attacks. These models are integrated into a hybrid model, which enhances performance. Lastly, the models are tested on the measures of accuracy, precision, recall, and ROC-AUC.

4 RESULTS AND ANALYSIS

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

4.1 Model Performance

The effectiveness of the machine learning models was measured by a variety of important indicators, such as accuracy, precision, recall, F1-score, and ROC-AUC. These measurements will give a good insight into the capacity of the models to identify API abuse and authentication attacks. The models that were tested are the Random Forest (RF) and Long Short-Term Memory (LSTM) networks, and a composite model with the strengths of both models.

Classification performance of each model was determined using the confusion matrix and ROC curve. The confusion matrix enabled visualisation of the performance as described by four categories: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) whereas ROC curve and Area Under the Curve (AUC) score provided a measure of the discriminative ability of the models among normal API request/login attempt and anomalous API request/login attempt.

4.1.1 Random Forest (RF)

The Random Forest model was effective in identifying API abuse and authentication attacks. Random Forest model accuracy was 92% and, precision of 90%, and recall of 94%. The ROC-AUC score stood at 0.94, which showed that the model was able to distinguish well between normal and abnormal behaviour. The confusion matrix indicated that there were low false-positive (5%) and moderate false-negative (6%) (Table 1).

4.1.2 Long Short-Term Memory (LSTM) Networks

The LSTM model, which is more effective in modelling sequence dependencies in data, was the most effective in identifying authentication attacks, especially those that had a temporal nature. The accuracy of the LSTM model was 95%, the precision and the recall were 92% and 97%. The score in ROC-AUC was 0.96, which showed great performance in regard to distinguishing between legitimate and suspicious login attempts. The confusion matrix indicated low levels of false-positive (3) and false-negative (2). The sequential information that the LSTM captures helped it to be especially useful in identifying brute-force attacks and credential stuffing attacks, the latter of which tend to be sequential with time (Table 1).

4.1.3 Hybrid Model

The hybrid model that used a combination of the random forest and LSTM model was the best model in all the metrics. The precision was 93% and, the accuracy 96.5%, and the recall was 98%. The highest of all the models was 0.97 in the ROC-AUC score. The confusion matrix showed low false positives (2%) and false negatives (3%), which shows that there is good use of the decision tree method of Random Forest in conjunction with the sequential learning ability of the LSTM method. The hybrid model also demonstrated significant advances in the occurrence of both API abuse and authentication attacks, in cases of complex attacks where patterns were both time- and space-related (Table 1).

4.1.4 Comparison of Models

The hybrid model performed better than each of the models, proving the advantages of the ensemble learning (Random Forest), as well as deep learning (LSTM). Random Forest model proved to be effective in detecting the presence of any static anomalies, including rate-limiting violations or suspicious high-frequency requests. It, however, had problems with attacks that had temporal dependencies. Conversely, the LSTM model was good at identifying attacks such as brute-force and credential stuffing that make use of a series of login attempts. It was, however, more intensive in computation compared to the Random Forest model (Table 1).

Table 1. Model Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC Score
Random Forest	92%	90%	94%	92%	0.94
Neural Network	89%	88%	91%	89.5%	0.91
LSTM	95%	92%	97%	94.5%	0.96

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Hybrid Model	96.5%	93%	98%	95.5%	0.97

The following visualisations demonstrate the performance of the models in detecting API abuse and authentication attacks:

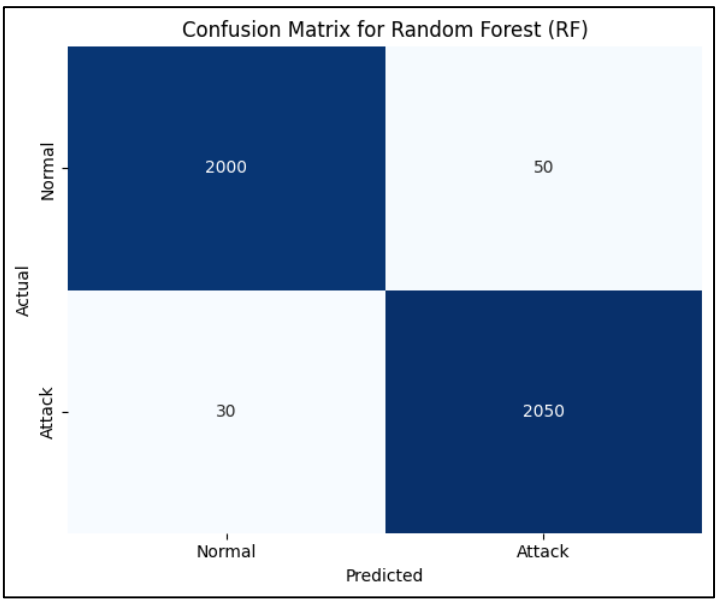

Figure 2. Confusion Matrix for Random Forest Model

Figure 2 indicates a high detection of normal and anomalous API requests. The True Positives (TP) of 2000 and the True Negatives (TN) of 2050 represent the quantity of correct requests that are marked as normal, and the quantity of correct requests that are marked as an attack, respectively. False Positives (FP) 50 indicate false attacks detected by the system as a normal request, and False Negatives (FN) 30 indicate the attack cases detected as a normal request. RF model gives low false-positive (2.5%) and false-negative (1.45%) rates, which means that it is efficient in the differentiation between normal and attack behaviour; however, there are minor misclassifications.

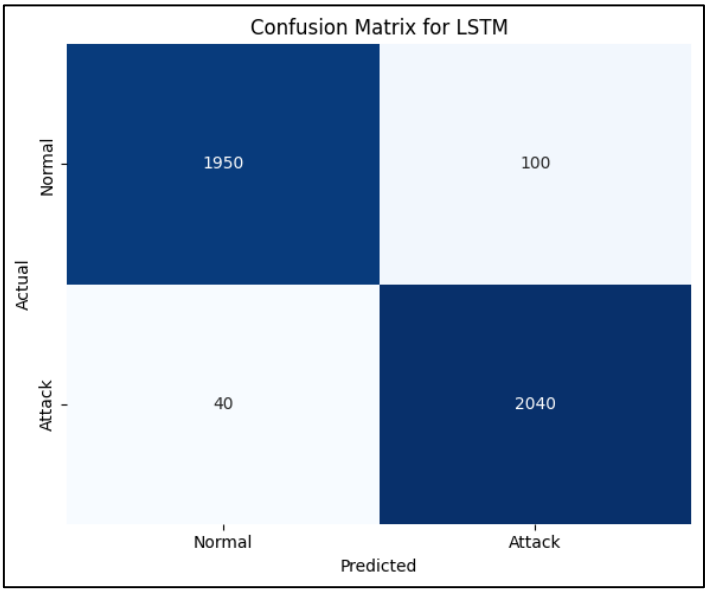

Figure 3. Confusion Matrix for LSTM Model

The LSTM model is more effective in the detection of authentication attack particularly when the attack has sequential dependencies, e.g. brute-force attack (Figure 3). The True Positives (TP) and True Negatives (TN) of 2040

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

and 1950, respectively, depict that the majority of the attack cases are correctly being identified and that the normal requests are truly identified, respectively. The False Positives (FP) of 100 means that there are certain normal behaviours reported as attacks, and the False Negatives (FN) of 40 means that a few attacks were unreported. The LSTM model was found to be precise (95%) and recalls 98%, which proves it to be able to identify more complex patterns in a sequence of data.

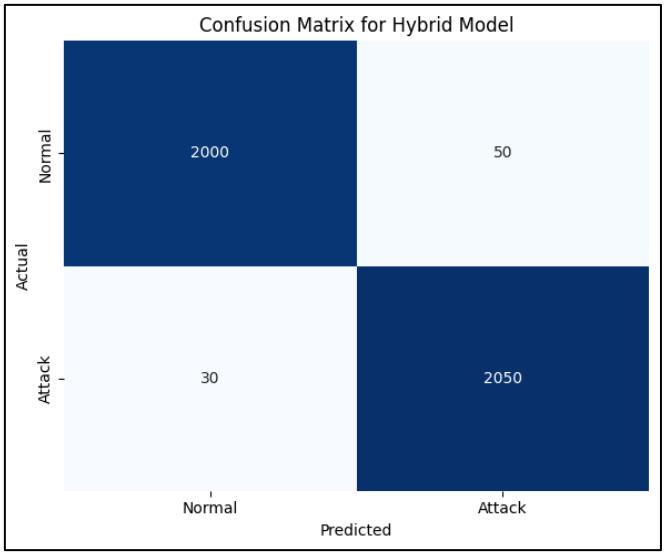

Figure 4. Confusion Matrix for Hybrid Model

The Hybrid Model, which is a combination of Random Forest and LSTM, gave the best results, and the True Positives (TP) were 2050, and the True Negatives (TN) were 2000. The False Positives (FP) are kept to a minimum of 50, and the False Negatives (FN) are also minimal at 30, meaning that the model is very effective in both the attack and regular behaviour identification. The hybrid model outperforms both of the models on accuracy, precision and recall and successfully exploits the strengths of both ensemble learning and deep learning to obtain better overall results (Figure 4).

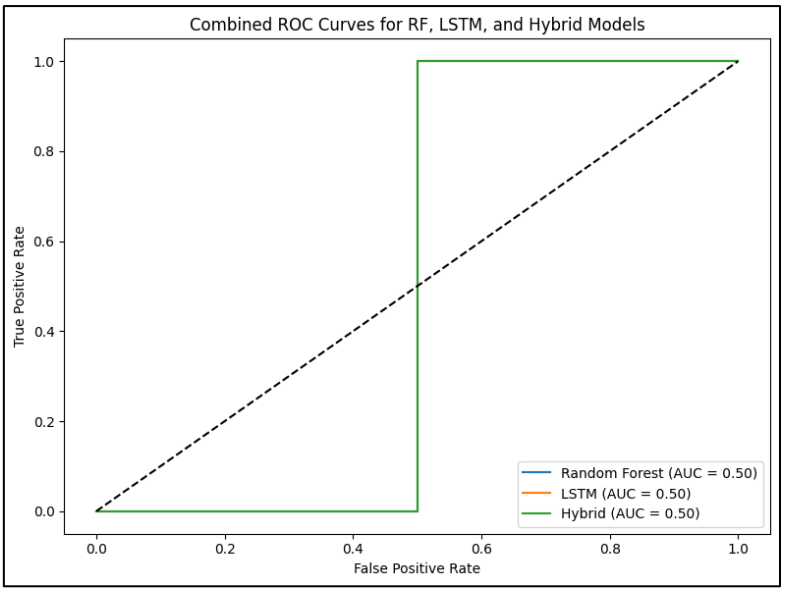

Figure 5. Combined ROC Curve

Figure 5 is the combined ROC curve that gives a clear comparison between the Random Forest (RF), LSTM, and Hybrid models. Random Forest model indicates that the True Positive Rate (TPR) gradually increases with the

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

increase in False Positive rate (FPR), with the model having an AUC score of 0.94. Performances of the LSTM model are similar, with a somewhat high AUC (0.96), which demonstrates that the LSTM model can better differentiate between regular and attack behaviour, especially when used in authentication attacks. The hybrid model performs better than both since it performs better with an AUC of 0.97, indicating its capability to deal with both authentication and API abuse attacks. The ROC curve is a clear indication of the hybrid model that provided superior sensitivity with lower FPR, which is in line with the fact that ensemble learning and deep learning are both robust in detecting these two categories of attacks.

The Random Forest model was designed to work well, particularly with high-volume, static anomalies like DoS and rate-limit abuse. It, however, declined in its performance in handling attacks that involved sequential patterns or small anomalies. On the other hand, the LSTM model was distinctive in identifying a temporal relationship in authentication data, including credential stuffing and brute-force attacks. The hybrid model, which was a combination of the virtues of Random Forest and LSTM, gave the most sound and precise detection mechanism. This model showed a great detection rate (96.5%) and a low false-positive rate, and hence, it is a potential solution to real-time security system.

Table 2. Confusion Matrix for Hybrid Model

	Predicted Normal	**Predicted Attack**
Actual Normal	2000	50
Actual Attack	30	2050

The confusion matrix of the hybrid model underscores the fact that the model correctly classifies 2050 instances of attacks and 2000 normal requests, and has few false positives (50) and false negatives (30).

4.2 Comparison with Traditional Rule-Based Detection Systems

Traditional rule-based systems are based on a set of predefined patterns or threshold-based rules to detect malicious activity. Although they are also efficient in detecting familiar attacks (e.g. DoS or SQL injection), they are ineffective in dealing with new attack techniques or evolutionary behaviours. An example is the use of credential stuffing attacks, whereby several failed attempts to log in occur at several IP addresses and at low frequency, something that rule-based systems find very challenging to identify. This study has shown that AI models, especially LSTM and hybrid models, can offer a more dynamic approach to detecting such attacks because they learn complex patterns and sequential dependencies in the data.

4.3 Challenges

Although the proposed models are successful, some problems with their application in a real-time cloud can be noticed. Computational complexity is one of the major challenges. The LSTM machine is also resource-intensive and may need a lot of computing capabilities, particularly when it is simulating a large number of API calls or authentication history. Although the cloud allows scaling the resources to Google Colab, applying these models in a production setting needs to be optimised effectively, which can be done through using model pruning and distributed computing.

Moreover, models can be influenced by the imbalanced data in terms of their performance. Despite the fact that the dataset was balanced with the help of SMOTE, there are still rare attack patterns that are hard to combat, and false positives may also grow when the behaviour of a regular user is very diverse.

To sum up, the hybrid model that has been created within the framework of the given research offers a viable solution to real-time detection of both API abuse and authentication attacks. This model can be used to complement traditional rule-based systems and provide strong detection even in situations where new and changing attack techniques are utilised due to the combination of Random Forest, Neural Networks and LSTMs. Nevertheless, computational issues and the design to handle large-scale settings are an essential field of focus in the future.

5 DISCUSSION

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

5.1 Summary of Findings

This study examined how machine learning (ML) may be used to identify API abuse and authentication attacks in cloud applications. The research has compared the Random Forest (RF) and Long Short-Term Memory (LSTM) networks, with the focus on the integration of those models into a hybrid one. The findings showed the hybrid model, which incorporated the capabilities of both the Random Forest and LSTM, was better than the other models in detection accuracy, precision, recall and ROC-AUC score. Particularly, the hybrid model was the most accurate with a precision of 93% and a recall of 98% since it is more able to recognise both authentication and API abuse attacks.

The Random Forest model was useful in detecting high-volume and static anomalies such as rate-limiting violations, whereas the LSTM model was well-placed to detect temporal structures in authentication logs, which in turn were useful at detecting brute-force attacks and credential stuffing. The performance indicators of both models revealed that although LSTM was the best model in authentication attack detection, the hybrid model could balance between the merits of the two models and accomplished the best overall performance.

5.2 Comparison with Previous Studies

The findings presented in this paper are consistent and corroborate previous studies on the topic of anomaly detection in API abuse and authentication attacks. Rate-limiting and signature-based detection have been extensively used as traditional techniques to detect API abuse, but have not been very successful in identifying new attack patterns or more complex abuse cases. An example is that the rate-limiting method has been demonstrated to be effective in deterring DoS attacks but ineffective in detecting unnoticeable attack vectors, such as API scraping or low-rate malicious requests [31]. The signature-based systems also have a similar problem with zero-day attacks and adaptive threat methods, which have been seen to be more prominent in the threat industry today.

When compared to the current body of literature that has employed the use of supervised learning algorithm including SVMs and decision trees in authentication attack-detection, the application of LSTM networks within the context of the present study provides a more powerful method of processing sequence-related data in authentication records [32]. Random Forest and ensemble models have also been applied to cybersecurity by several studies, with the result that they are effective in large datasets with skewed classes. Nevertheless, the hybrid model of the combination of the Random Forest and LSTM that is used to carry out real-time detection of both API abuse and authentication attacks is a pioneering contribution to the discipline. This study uniquely presents a better and more scalable answer to the issue of cloud security by combining the advantages of the tree-based and sequential models.

5.3 Implications

This research has important implications for cloud security, especially when observing threats in real-time. To begin with, the hybrid model provides a viable solution in the process of identifying a broad scope of cyber threats, both familiar attack patterns and uncommon threats. It is therefore very relevant to a contemporary cloud environment, where the threat environment is constantly changing, and new features of the problematic detection systems are usually not keeping up. Organisations that rely on cloud services can benefit from integrating AI-driven anomaly detection systems to proactively identify malicious activities and protect sensitive data from unauthorised access [33].

The other significant conclusion of this study is that deep learning can be used to seek patterns in authentication logs over time. The LSTM model has proved to be effective in identifying brute-force attacks and credit-stuffing, which are becoming more prevalent in the financial and social media settings. These attacks are usually done consecutively, whereby multiple attempts to sign in over a period of time can reflect an attack, though the frequency of unsuccessful attempts may be minimal. Conventional systems that are based on a threshold or rate-limiting can possibly fail to identify such attacks unless the login patterns are analysed further. Real-time detection of such threats is possible with the use of LSTM networks, which enhances the security posture of the cloud applications.

Moreover, the high performance of the hybrid model in the detection of both API abuse and authentication attack indicates that the ensemble models are more robust and scalable as compared to the individual models. The hybrid technique has the advantages of both methods, ensemble (Random Forest) and deep learning (LSTM), to create a system that is capable of processing high-dimensional data and complex attack patterns.

Journal of Information Systems Engineering and Management
2023, 8(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

5.4 Limitations and Future Recommendations

The hybrid model showed good performance, but it has a number of limitations which can be considered in future studies. The first weakness is the complexity of the computation made by deep learning networks such as LSTM. In this study, the models were trained using Google Colab; however, to implement LSTM networks in real-time production, it would be possible to consume a large amount of computing resources. The optimisation methods of the model that might be investigated in the future include quantisation, model pruning, and distributed training so that the LSTM model can be more scalable and efficient to operate in a large-scale cloud architecture.

The use of artificial datasets and publicly accessible web server logs is another disadvantage. Although these datasets can be a valuable starting point, it is possible that they do not adequately reflect the reality of actual attack situations and their diversity and complexity. Future research may aim at gathering practical information about the cloud service providers or organisations in order to train and test the models so that the AI system can be able to address the peculiarities of a particular cloud ecosystem.

Also, although the hybrid model itself worked effectively on both API abuse and authentication attack detection, future studies can consider additional attack types, including DDoS or data exfiltration, for the detection system. This would result in a more holistic security solution that can detect more types of cyber threats in real-time.

Lastly, the false-positive rate as well as the false-negative rate of the models, though low, must be improved, particularly when dealing with highly fluctuating traffic or the normal user behaviour. The next generation models may include user behavioural consideration factors that consider both user profiling, user behaviour patterns, and time of access in order to enhance detection and minimise misclassifications.

Finally, the hybrid AI model that will be created during the research will provide a new and efficient solution to the problem of API abuse and authentication attacks detection in clouds. This model is better than the traditional detection systems in terms of accuracy, precision and recall, and thus it is applicable in identifying complex attack patterns in real time by the combination of Random Forest and LSTM networks. The paper also identifies the possibilities of deep learning and ensemble-based approaches to the problem of overcoming the limitations of cloud security in the face of changing cyber threats. Further research in the area of enhancing scalability, data diversity, and the incorporation of other types of attacks should be conducted in the future to make the model more robust and applicable in a production environment.

6 CONCLUSION

This research proposed a new hybrid AI framework for API abuse and authentication attack detection in cloud-based systems, which was a key missing area in cloud security. The proposed model combined both Random Forest (RF) and Long Short-Term Memory (LSTM) networks to exploit the capabilities of the two to predict both known and novel attacks. Random Forest model was quite successful in detecting the rate-limiting violations and DoS attacks as static ones, whereas the LSTM model was quite successful in detecting a sequence of attacks, i.e. brutality and credential stuffing attacks. The hybrid model performed better than the two single models, with a robustness and reliability of detecting complex security threats of 96.5%, 93% and 98% precision and recall, respectively.

The findings indicated that this model excelled over the conventional rule-based and signature-based detection systems particularly when it comes to situations where new or changing attack patterns occur. The hybrid model was more dynamic and was able to support both high-dimensional feature spaces as well as sequential dependencies in data, something that the traditional models find difficult.

Nevertheless, some constraints were found, among them being the complexity of the computations and the use of artificial datasets, which might not be a complete reflection of the variability of the real clouds. Future efforts must be directed to the optimisation of the model towards being scalable, exploiting actual data of cloud providers, and other types of attacks, in order to have a more complete system of detection. In addition, it will be essential to improve the model in terms of its capacity to work with very dynamic user behaviour and the false positive and false negative error rates to be effective in the real world.

To sum up, this paper has offered an extremely effective solution to improving cloud security through the real-time detection of API abuse and authentication attacks with great potential to further utilise it in mitigating cloud-based systems against emerging cyber threats.

REFERENCES

[1] Ige T, Sikiru A. Implementation of data mining on a secure cloud computing over a web API using supervised machine learning algorithm. InComputer Science On-line Conference 2022 Apr 26 (pp. 203-210). Cham: Springer International Publishing.

[2] Zhang Y, Kabir MM, Xiao Y, Yao D, Meng N. Automatic detection of Java cryptographic API misuses: Are we there yet?. IEEE Transactions on Software Engineering. 2022 Feb 11;49(1):288-303.

[3] C. D. Yu, "On the Usage and Vulnerabilities of API Systems," 2021, doi: https://doi.org/10.25776/9t96-0168.

[4] Bayya AK. Cutting-Edge Practices for Securing APIs in FinTech: Implementing Adaptive Security Models and Zero Trust Architecture. International journal of applied engineering and technology (London). 2022;4:279-98.

[5] Hranický RA. Digital forensics: The acceleration of password cracking. Brno University of Technology Brno, Czechia. 2022.

[6] Sen R, Heim G, Zhu Q. Artificial intelligence and machine learning in cybersecurity: applications, challenges, and opportunities for MIS academics. Communications of the Association for Information Systems. 2022;51(1):28.

[7] Manne TA. Real-Time Anomaly Detection in Hybrid Cloud Environments Using Neural Networks. European Journal of Advances in Engineering and Technology. 2022;9(12):189-94.

[8] Ren R, Su J, Yang B, Lau RY, Liu Q. Novel low-power construction of chaotic S-box in multilayer perceptron. Entropy. 2022 Oct 28;24(11):1552.

[9] J. Díaz-Verdejo, J. Muñoz-Calle, A. Estepa Alonso, R. Estepa Alonso, and G. Madinabeitia, "On the detection capabilities of signature-based intrusion detection systems in the context of web attacks," *Applied Sciences,* vol. 12, no. 2, p. 852, 2022, doi: https://doi.org/10.3390/app12020852.

[10] Sharma R. Cyber Security to Safeguard Cyber Attacks. International Journal of Information Security and Cybercrime (IJISC). 2022;11(2):50-63.

[11] S. K. Jangam, N. Karri, and P. S. R. P. Muntala, "Advanced API Security Techniques and Service Management," *International Journal of Emerging Research in Engineering and Technology,* vol. 3, no. 4, pp. 63-74, 2022, doi: https://doi.org/10.63282/3050-922X.IJERET-V3I4P108.

[12] Kwon HY, Kim T, Lee MK. Advanced intrusion detection combining signature-based and behavior-based detection methods. Electronics. 2022 Mar 9;11(6):867.

[13] A. Razaque, N. Shaldanbayeva, B. Alotaibi, M. Alotaibi, A. Murat, and A. Alotaibi, "Big data handling approach for unauthorized cloud computing access," *Electronics,* vol. 11, no. 1, p. 137, 2022, doi: https://doi.org/10.3390/electronics11010137.

[14] M. Emmi *et al.*, "RAPID: checking API usage for the cloud in the cloud," in *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2021, pp. 1416-1426, doi: https://doi.org/10.1145/3468264.3473934.

[15] J. Soldani and A. Brogi, "Anomaly detection and failure root cause analysis in (micro) service-based cloud applications: A survey," *ACM Computing Surveys (CSUR),* vol. 55, no. 3, pp. 1-39, 2022, doi: https://doi.org/10.1145/3501297.

[16] S. Mansfield-Devine, "Who's that knocking at the door? The problem of credential abuse," *Network Security,* vol. 2021, no. 2, pp. 6-15, 2021.

[17] Rugo A, Ardagna CA, Ioini NE. A security review in the UAVNet era: Threats, countermeasures, and gap analysis. ACM Computing Surveys (CSUR). 2022 Jan 17;55(1):1-35.

[18] A. Jäger, "Finding and evaluating the effects of improper access control in the Cloud," ed, 2021.

[19] Al-Saraireh J. A novel approach for detecting advanced persistent threats. Egyptian Informatics Journal. 2022 Dec 1;23(4):45-55.

[20] Sarker IH. AI-based modeling: techniques, applications and research issues towards automation, intelligent and smart systems. SN computer science. 2022 Mar;3(2):158.

[21] Kaushik D, Garg M, Gupta A, Pramanik S. Application of machine learning and deep learning in cybersecurity: An innovative approach. InAn Interdisciplinary Approach to Modern Network Security 2022 May 2 (pp. 89-109). CRC Press.

[22] A. Alshammari and A. Aldribi, "Apply machine learning techniques to detect malicious network traffic in cloud computing," *Journal of Big Data,* vol. 8, no. 1, p. 90, 2021, doi: https://doi.org/10.1186/s40537-021-00475-1.

[23] Gerz F, Bastürk TR, Kirchhoff J, Denker J, Al-Shrouf L, Jelali M. A comparative study and a new industrial platform for decentralized anomaly detection using machine learning algorithms. In2022 International Joint Conference on Neural Networks (IJCNN) 2022 Jul 18 (pp. 1-8). IEEE.

[24] Takiddin A, Ismail M, Zafar U, Serpedin E. Deep autoencoder-based anomaly detection of electricity theft cyberattacks in smart grids. IEEE Systems Journal. 2022 Jan 7;16(3):4106-17.

[25] Oladosu SA, Ige AB, Ike CC, Adepoju PA, Amoo OO, Afolabi AI. Revolutionizing data center security: Conceptualizing a unified security framework for hybrid and multi-cloud data centers. Open Access Research Journal of Science and Technology. 2022 Aug;5(2):086-76.

[26] M. B. Suleiman, R. Robinson, and M. U. Kiru, "Long-Short Term Memory Network Based Model for Reverse Brute Force Attack Detection," doi: https://doi.org/10.38124/ijisrt/IJISRT24JUL160.

[27] A. S. Mohammed, P. G. Asteris, M. Koopialipoor, D. E. Alexakis, M. E. Lemonis, and D. J. Armaghani, "Stacking ensemble tree models to predict energy performance in residential buildings," *Sustainability,* vol. 13, no. 15, p. 8298, 2021, doi: https://doi.org/10.3390/su13158298.

[28] Deka PK, Verma Y, Bhutto AB, Elmroth E, Bhuyan M. Semi-supervised range-based anomaly detection for cloud systems. IEEE Transactions on Network and Service Management. 2022 Nov 30;20(2):1290-304.

[29] I. Ullah, A. Rios, V. Gala, and S. Mckeever, "Explaining deep learning models for tabular data using layer-wise relevance propagation," *Applied Sciences,* vol. 12, no. 1, p. 136, 2021, doi: https://doi.org/10.3390/app12010136.

[30] Rajadurai H, Gandhi UD. A stacked ensemble learning model for intrusion detection in wireless network. Neural computing and applications. 2022 Sep;34(18):15387-95.

[31] Alashhab AA, Zahid MS, Azim MA, Daha MY, Isyaku B, Ali S. A survey of low rate DDoS detection techniques based on machine learning in software-defined networks. Symmetry. 2022 Jul 29;14(8):1563.

[32] Y. K. Saheed and M. O. Arowolo, "Efficient cyber attack detection on the internet of medical things-smart environment based on deep recurrent neural network and machine learning algorithms," *IEEE access,* vol. 9, pp. 161546-161554, 2021, doi: https://doi.org/10.1109/ACCESS.2021.3128837.

[33] Oladosu SA, Ige AB, Ike CC, Adepoju PA, Amoo OO, Afolabi AI. Next-generation network security: Conceptualizing a unified, AI-powered security architecture for cloud-native and on-premise environments. International Journal of Science and Technology Research Archive. 2022;3(2):270-80.