International Journal of Basic and Applied Sciences, 14 (2) (2025) 645-650

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS
https://doi.org/10.14419/3a4djt23
Research paper

Identifying Cyberattacks in Cloud Computing Service Frameworks Through Correlation-Driven Feature

Smitha G. V. [1] *, Dr. Samitha Khaiyum [2], Yashaswini D. S. [3], Yadhunandan G. N. [3], Yarramreddy Sai Kumar Reddy [3], Yogeshwar S [3]

[1] Assistant Professor, Research Scholar, Department of MCA , Dayananda Sagar College of Engineering , Bangalore
[2] Professor Department of MCA Dayananda Sagar College of Engineering , Bangalore
[3] Department of MCA Dayananda Sagar College of Engineering , Bangalore
**Corresponding author E-mail: smitha-mca@dayanandasagar.edu*

Received: May 27, 2025, Accepted: June 24, 2025, Published: July 7, 2025

Abstract

Cloud computing infrastructures are often targeted by hacktivists due to their relatively insecure service and delivery paradigms. This article addresses common attack vectors, such as ransomware, insider threats, API breaches, and data leaks, that, if ignored, might jeopardize sensitive data and interfere with important work. We provide a new method based on correlation-based feature selection for identifying fraudulent activities in cloud systems. Using Pearson correlation analysis, we drastically cut down on redundant features on the security dataset, yielding promising results for threat identification. Our feature selection method's effectiveness is empirically supported, and we demonstrate that the classification algorithms perform better when using the improved tool of the optimized data set than when using the original feature set. In this research, a unique approach to intelligent data pre-processing for cloud security enhancement is presented.

Keywords: *Cloud Security; Cyberattack Detection; Feature Engineering, DDoS Defense; and Pearson Correlation Analysis.*

1. Introduction

End users obtain resources from cloud computing in the form of computer resources as a service over the Internet, and this approach has several uses, e.g., shorter application development times, a pay-per-use pricing model, and shared service costs. Since the technology is easily scalable, you can quickly modify how much service you have in responding to fluctuations in demand. Further, it is possible to add or remove resources instantly, due to elasticity. Secure remote access, higher system reliability, lower DR costs, and fast service restoration are just a few positive by-products. Businesses are now creating volumes of data that they could not possibly manage without cloud computing, whereas Industry 4.0 technologies are growing across many industries. From IoT devices and municipalities to smart factories and drone swarms, the world of today requires the same complex cloud platform to operate such critical infrastructure. While the adoption of the cloud platform is increasingly favoured, the increased adoption of cloud computing services technology is held back by security concerns. Security (74%) remains the number one issue for business, and reluctance is the result for many to go all in [1].

Our work presents a new approach for detecting online threats from cloud systems. For virtualized environments, we proposed a new feature selection algorithm. We first analyse the relations between different system features in detail, pointing out crucial links using the method of Pearson's correlation. Our experiments indicate that, as compared to the source dataset, when the KNN algorithm is applied to the optimized feature set, the detection performance is higher. We suggest that the processed data reflects the effectiveness of our correlation-based feature selection in improving cloud security. This method is precise and fast and has achieved remarkable performance in identifying advanced threats from cloud infrastructures.

2. Relevant work

Cloud computing security has been extensively researched in the past few years, as cyber threats in virtual environments have become more prevalent. Early research like Alguliev and Abdullayeva [1] examined overall security issues in cloud computing, whereas the Cloud Security Alliance's "Treacherous Twelve" report [2] outlined dominant threats to cloud platforms, including data loss and DDoS attacks.

Detection of distributed denial-of-service (DDoS) attacks on cloud platforms has been addressed by various machine learning and feature engineering models. Aamir and Zaidi [3] promoted a hybrid approach combining feature engineering with machine learning for DDoS detection. Similarly, Zi et al. [4] presented an adaptive clustering model with feature ranking to improve the accuracy of DDoS detection. Deka et al. [6] concentrated on ranked learning methods to determine features most appropriate for the identification of DDoS attacks, enhancing classifier performance in cloud datasets.

The use of modern datasets like CSE-CIC-IDS2018 [10] and HTTP CSIC 2010 [11] has enabled benchmarking of new detection techniques. Researchers like Moustafa et al. [14] and Singh et al. [15] explored big data analytics and agile cloud security models to strengthen intrusion detection systems (IDS). AI-driven and deep learning approaches for detecting complex threats, including zero-day and DDoS attacks, were proposed by Zhang and Wu [16] and Shafiq et al. [17]. The increasing reliance on multi-cloud environments and AI-driven attacks has led to calls for innovative detection strategies. Recent works, such as Sahu et al. [19], demonstrated the value of ensemble classifiers with feature selection, while Abdullayeva [7] applied softmax regression with autoencoders for advanced persistent threat detection. These studies collectively highlight the importance of selecting optimal feature subsets, balancing detection accuracy with computational efficiency, and addressing emerging attack vectors in modern cloud infrastructures.

3. Cloud computing service delivery models

As demonstrated in fig. 1, the three primary cloud service paradigms are Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS).

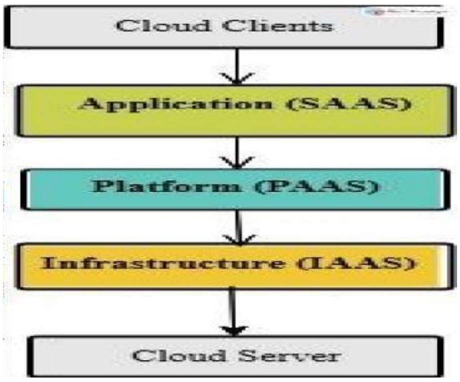

Fig. 1: Cloud Service Model.

SaaS is a very popular model of cloud computing in which the service providers host and make applications available to users. Although SaaS has undeniable benefits in terms of price reductions and convenience of use, it also creates major vulnerabilities in regards to system security. The most typical issues are information leakage, software vulnerabilities, unauthorized intrusion, and network-level risks. Besides, such problems as data isolation, poor identity management, and virtualization-related risks also make cloud security more complicated. Cyberattacks that target SaaS environments include Cross-Site Scripting (XSS), SQL injection, Man-in-the-Middle (MITM) attacks, port scanning, IP spoofing, packet sniffing, Distributed Denial-of-Service (DDoS), and other HTTP-based attacks. To accomplish strong cloud security and properly identify such cyber threats, we used the dataset of HTTP CSIC 2010 as a common benchmark. The main contributions of our research consist of emphasizing the significance of employing the K-Nearest Neighbors (KNN) model along with the correlation-based feature selection to increase the detection accuracy. Our approach has the systematic identification of the most relevant attributes, noise reduction, and optimization of the classification performance, unlike other traditional models, which indiscriminately process all available features. Our approach outperforms the accuracy with minimal computational cost, which is essential in real-time cloud security solutions due to emphasis on important features using correlation analysis and the ability of KNN to classify data using distance measure as a distance-based classifier.

4. Cloud computing landscape

Cloud services are particularly vulnerable to network intrusion, which may jeopardize the core applications, lower the bandwidth, hamper the operations, and pace resources. One of the most severe of these threats is the Distributed Denial-of-Service (DDoS) attacks. Botnets are also used by attackers to saturate target servers with traffic. Often, the traffic will be masked with thousands of spoof IP addresses. Due to its scattered state, it is hard to mitigate because the traditional security measures cannot always differentiate between a malicious and a legitimate request. Overall, it is possible to distinguish two types of DDoS attacks: Network/Transport Layer (Layers 34) attacks. Through these attacks, routers are overloaded, bandwidth is flooded, or network resources are depleted to affect the infrastructure. The weaknesses in protocols that attackers utilize include: UDP Floods: UDP packets in huge amounts are sent to server ports. TCP SYN Floods: Hackers abuse the connection establishment procedures to waste connection resources. ICMP Floods: Too many ping requests flood and impact services. Layer 7 Application Layer Attacks Further sophisticated than network-layer attacks, these attacks target application-level resources such as CPU, memory, and storage by means of application vulnerabilities. The nature of these threats is constantly changing will require a high level of detection mechanisms that must analyse traffic patterns to reduce false positives. In that regard, we use K-Nearest Neighbours (KNN) as a base classifier to identify and label attack traffic with high legitimacy in our research. Backed with correlation-based feature selection, the KNN model allows accurate detection of malicious activities since it classifies the traffic according to the distance measures in the feature space. This dynamic technique enables dynamic filtering of suspect traffic and strengthening protocol-based defensive measures, and is therefore quite suitable for real-time protection in a cloud computing environment.

5. Criteria for correlation

We have inputs xij (features) and Yk (target values) in the applications of machine learning, where x indicates the sample instances and ye indicates the corresponding class labels. To determine important features, we use the Pearson correlation approach, which is a powerful statistical algorithm to measure the linear relations among variables.
This correlation can be formally defined as:

$$R(i) = \frac{cov(x_i, y)}{}$$

$$\overline{\sqrt{var(x_i) \cdot var(y)}}$$

- x refers to the i- th feature.
- Y is the output class label Vector.
- cov() refers to the covariance function.
- var() denotes the variance function.

These correlation scores are used in our study to help select the relevant features before the application of the K-Nearest Neighbors (KNN) model. It makes sure that the most informative attributes are only used to distance-based classification, which improves both the accuracy and computational efficiency.

6. Experiments

The correlation matrix visualization (Figure 2) demonstrates important relationships among the features in the HTTP CSIC 2010 dataset. The essence of effective feature selection based on correlation analysis suggests that the optimal feature set must be as non-redundant as possible, and at the same time, have a good predictive correlation with target variables. These two-fold criteria warrant that the features picked are those that offer the most information gain with the least overlap. To use this idea, we combine two analysis methods that complement each other. A thorough brute-force analysis is performed first, searching through the space of all feature subsets to find the most suggestive ones. Although this brute force approach is sure to provide a global evaluation, it is computationally expensive in high dimensions. Second, a cluster-based analysis identifies the natural grouping of strongly correlated features in the data. Usually, one identifies clusters of three to seven highly correlated features. The dimensionality is reduced, thus allowing predictive performance to be maintained by simply picking a single representative feature per cluster. The experiments on the CSE-CIC-IDS2018 and HTTP CSIC 2010 datasets prove the efficiency of this methodology. Interestingly, the method logged good performance on the HTTP CSIC 2010 dataset that consists of 71,484 instances distributed over 21 attributes in the binary classification format (0 abnormal traffic, 1 normal traffic). The correlation analysis was effective at removing redundant features and keeping the most important ones to improve the distance-based classification performance of KNN in separating malicious and legitimate traffic.

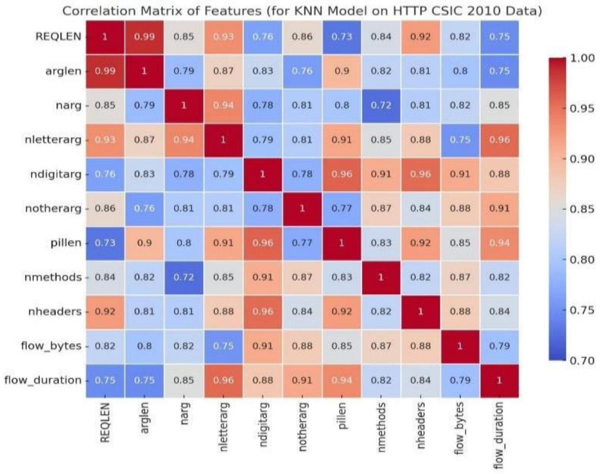

Fig. 2: Correlation Matrix of the Features of the HTTP CSIC 2010 Dataset.

Rows and columns in this matrix correspond to the features of the dataset, and the correlation between a given pair of attributes is displayed in the cell. The intensity of colors shows the strength of association among features. As it should be, diagonal elements correspond to perfect correlation of 1.0, which is the relation of each feature to itself. Weaker correlations between pairs of features are shown by lighter colored cells. Figure 2 indicates that correlation values within the range of 0-0.75 indicate significant inter-feature relationships. A value of 0.75 was set to determine and screen correlated features. As an illustration, the attributes like REQLEN are highly interrelated with several other attributes, which highlights its significance in the optimal feature set employed by the KNN classifier.

Table 1: Feature Name & Average Correlation Score

Feature Name	Average Correlation Score
REQLEN	0.899
arglen	0.879
narg	0.859
nletterarg	0.890
ndigitarg	0.872
notherarg	0.875
pillen	0.894
nmethods	0.871
nheaders	0.884
flow_bytes	0.886
flow_duration	0.880

6.1. Description of the features

- The entire time taken during which the request was sent to the server is referred to as the Request Length (REQLEN).
- The argument length (arglen) is the length of the arguments submitted in the HTTP request.
- The number of arguments (narg) contained in the request is supplied.

- The total amount of arguments, including letters, is referred to as the number of letter arguments (nletterarg).
- The number of arguments comprised of numeric characters is called the number of Digit Arguments (ndigitarg).
- The number of arguments that include special characters/symbols is referred to as the number of other arguments (notherarg).
- The payload length (pillen) is the length of the HTTP request payload being sent to the server.
- The number of methods (nmethods) indicates the number of different HTTP methods (e.g., GET and POST) that are used during the session.
- The size of the request in terms of the number of HTTP headers is represented by nheaders.
- The number of bytes transferred throughout the HTTP request-response loop is called the flow bytes (flow_bytes).
- Flow Duration (flow_duration): The duration of time the client and server are communicating in total.

Table 2: Results of the Classification of the Approach on the "Http Csic 2010" Dataset

Methods	Original Dataset ROC-AUC	Newly Dataset AUC	Formed ROC-
Random Forest	Train set: 0.9263 Test set: 0.9240	Train set: 0.9452 Test set: 0.9437	
Logistic Regression	Train set: 0.9347 Test set: 0.9329	Train set: 0.9602 Test set: 0.9591	
KNN (New)	Train set: 0.9225 Test set: 0.9210	Train set: 0.9485 Test set: 0.9462	

The experimental results demonstrate significant gains when the original dataset is compared to correlation-optimized data. Classification of the unprocessed data using Random Forest gave a ROC score of 0.9263 on the training set, which improved to 0.9452 when correlation-processed features were used. Likewise, the performance of the Logistic Regression increased in the training set: Logistic Regression performance: 0.9347-0.9602.

Next, we introduce the K-Nearest Neighbors (KNN), our correlation-processed dataset, boosted ROC-AUC of KNN in both the training and test sets, to 0.9485 and 0.9210, respectively. Such gains verify the usefulness of our feature selection strategy in improving the detection performance across classifiers.

Fig. 3 shows the correlation matrix of features of the CSE-CIC-IDS2018 dataset, which was analyzed in our work. The matrix reveals the interdependencies among the network traffic features, with the correlation values mostly located between 0.75 and 1.0. Features like Dst Port, TotLen Fwd Pkts, and Fwd Pkt Len Max recorded strong correlations.

A threshold of 0.75 was used to eliminate 24 correlated features and diminish the size of the dataset to 55 features. This dimensionality reduction enhanced the computational performance as well as the classification accuracy of the KNN model.

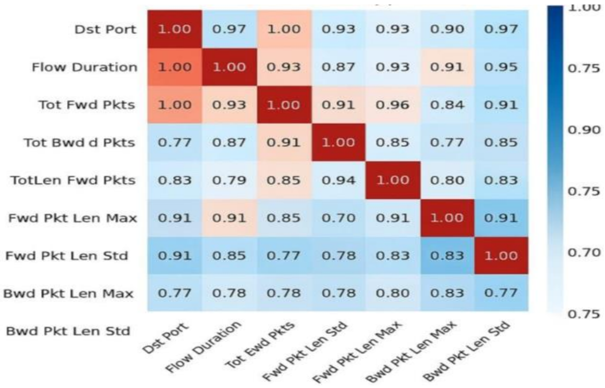

Fig. 3: Correlation Matrix between the 'CSE-CIC ID2018' Dataset's Features (KNN Approach).

A threshold of 0.75 was used to eliminate 24 correlated features and diminish the size of the dataset to 55 features. This dimensionality reduction enhanced the computational performance as well as the classification accuracy of the KNN model.

Table 3: The "DST Ports'" Most Closely Related Specifications

DST PORT	0.907733
Flow Duration	0.872735
Tot Fwd Pkts	0.936844
Tot Bwd Pkts	0.785975
Totlen Fwd Pkts	0.933529
TotLen Bwd Pkts	0.887467
Fwd Pkt Len Max	0.836548
Fwd Pkt Len Std	0.788872
Bwd Pkt Len Max	0.968219
Bwd Pkt Len Std	0.781297

6.2. Elementary flow metrics

- DST PORT: Specifies the service being connected to of a communication by giving the destination port number. Flow Duration: Indicates the overall connection time, and it is usually in microseconds.
- Tot Fwd Pkts: Displays the total number of packets that have been forwarded by the source to the destination.
- Tot Bwd Pkts: Refers to the total number of packets that were transmitted as a response by the destination to the source.
- Packet Volume and Size Tracking TotLen Fwd Pkts: Counts the total length (in bits) of packets forwarded or transmitted.
- TotLen Bwd Pkts: This is the total number (in bytes) of the packets that have been sent in reverse (destination to the source).
- Max Fwd Pkt Len / Max Bwd Pkt Len: Note the largest packet size in the forward and backward directions, respectively.

- Pkt Len Max: Captures the largest packet size seen in both directions.
- Statistical Distributions Fwd Pkt Len Std / Bwd Pkt Len Std: Test the uniformity or range of forward and backward packet sizes.
- Pkt Len Std: Shows the variance of packet lengths of the whole flow.
- Pkt Len Mean: Indicates the mean packet size within the flow.
- Timing Descriptions Flow Bytes/s: This is the speed of the communication transmission, which is calculated in bytes per second.
- Fwd IAT Tot / Fwd IAT Max: Timing patterns of captures in the forward direction, forward total, and maximum inter-arrival time.
- Flow IAT Max: The maximum inter-packet time interval of the flow. Protocol-Specific Elements Fwd Header Len / Bwd Header Len: Evaluate the protocol overheads by determining the length of the forward and backward packet headers.
- Subflow Measures: Offer a level of detail by breaking flows into subflows to gain insights into minute details.
- Init Bwd Win Byts: This is the initial window size for backwards flow control.
- Data Packets: Occument A packet that carries information, especially in file transfer protocols.

These clearly defined metrics offer a complete network behavior analysis ground. They facilitate accurate distinction between normal activities and possible security abnormalities as they capture volume, time of day, size distribution, and protocol-specific features. Such a feature set reinforces traffic analysis and threat identification efforts, even our KNN-based model.

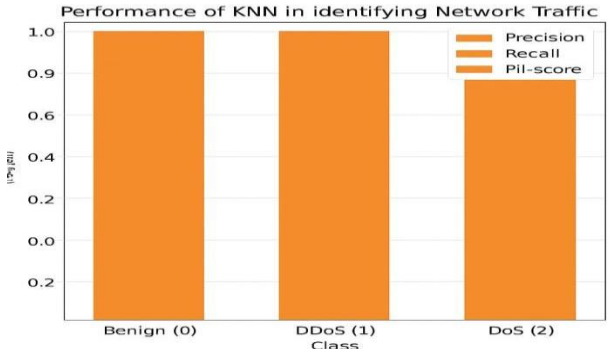

Fig. 4: Dst Port and Flow Duration Distribution of KNN Classification on CSE-CIC-IDS2018 Dataset.

Table 4: Classification Performance of the KNN Model on CSE-CIC-IDS2018 Dataset

Method	Dataset Classes	Precision	Recall
KNN	Benign (0)	0.9935	0.9951
	DDoS (1)	0.9958	0.9942
	DoS (2)	0.9921	0.9910

The suitability of the suggested algorithm in detecting traffic in a network was rigorously tested with three key performance measures, namely precision, recall, and F1-score. As shown in Table 4, the model achieved high precision in all three traffic types, namely benign (Class 0), DDoS (Class 1), and DoS (Class 2).

With the correlation-based feature selection and K-Nearest Neighbors (KNN) as a base algorithm, our method showed high precision, recall, and F1-scores across all classes, normal traffic (Class 0), and attack scenarios (Classes 1 and 2), with minimal variations. The KNN model returned the best precision of 0.9935, recall of 0.9951, and the F1-scores of 0.9943, indicating a well-balanced performance on all the categories. This implies that the model not only reduces errors, including failures to identify an attack or false positives, but it also keeps a proper separation between regular and malicious flows. It is worth noting that the technique was very efficient in identifying any DDoS attacks, which confirmed that it could manage multifaceted and massive threats. These results suggest the applicability of our approach to practical cyber defense: our technique achieves high detection accuracy with low false positive and negative rates and can be applied in a real-time intrusion detection system (IDS) to settings where fast and reliable performance is critical.

7. Limitations and future scope

Although the suggested framework integrating correlation-based feature selection with K-nearest neighbors (KNN) shows strong detection performance over a wide range of attack types in cloud environments, it is not flawless. The testing was mainly on balanced and clean datasets (CSE-CIC-IDS2018 and HTTP CSIC 2010), which are perhaps not representative of real-world cloud traffic, with imbalance and noise being typical. The existing method is centered on linear correlations (Pearson correlation) for feature selection, which, while efficient in eliminating redundancy, can ignore non-linear relationships between features that may improve detection of sophisticated threats like zero-day or stealthy attacks.

Second, while the framework saves computational expense, the incremental improvements in metrics like ROC-AUC, especially for certain classifiers, need formal statistical testing to

verify significance. In the future, the framework may be extended to support imbalanced and
streaming data across multi-cloud or hybrid-cloud settings, including non-linear correlation

Methods or mutual information-based selection to identify deeper relationships between features and investigate integrating KNN with ensemble or hybrid deep models for better flexibility. In addition, implementing the system on actual cloud infrastructures and subjecting it to adversarial and dynamic attack patterns to test its resilience will be crucial steps toward effective, real-time intrusion detection systems.

8. Conclusion

To boost cloud security against such dynamic cyber-threats, this paper proposes an improved attack detection model that integrates correlation-based feature selection with the K-Nearest Neighbors (KNN) classifier to achieve improved cloud security. By using Pearson correlation analysis to select and keep the most informative features of high-dimensional network data, our method can effectively increase the detection accuracy with a smaller computational burden.

The engineered feature sets outperformed the earlier approaches when tested with KNN on two benchmark datasets, CSE-CIC-IDS2018 and HTTP CSIC 2010. Experimental evidence shows that our model can produce a high level of classification accuracy in a variety of different attack scenarios, such as DDoS, DoS, and normal traffic.

These results confirm the suitability of our approach to real-life cloud security practices. Besides improving the detection, the model also provides efficiency and hence would be very applicable in real-time intrusion detection systems where reliability and fast response time are of great importance.

References

[1] R.M. Alguliev, F.C. Abdullayeva, “An investigation and analysis of security problems of the cloud computing,” Problems of Information Technology, 2013, №1(7), pp. 3-14.

[2] ‘The Treacherous Twelve’ Cloud Computing Top Threats in 2016, https://cloudsecurityalliance.org/artifacts/thetreacherous- twelvecloud- computing-topthreats-in-2016/

[3] M. Aamir, S.M. Zaidi, “Detecting DDoS attacks using feature engineering, machine learning, the framework, and performance assessment The International Journal of Information Security, 2019, vol. 18, pp. 761- 785. https://doi.org/10.1007/s10207-019-00434-1.

[4] IEEE 4-th International Conference on Network and System Security, "Adaptive Clustering with Feature Ranking for DDoS Attacks Detection," L. Zi, J. Yearwood, X.W. Wu, 2010, pp. 282286.

[5] G. Chandrashekar, F. Sahin, “A survey on feature selection methods,” Computers and Electrical Engineering, 2014, vol. 40, no. 1, pp. 16- 28. https://doi.org/10.1016/j.compeleceng.2013.11.024.

[6] R.K. Deka, D.K. Bhattacharya, J.K. Kalita, “Active utilizing ranked learning to identify DDoS attacks features,” Computer Communications, 2019, vol. 145, pp. 203-222. https://doi.org/10.1016/j.comcom.2019.06.010.

[7] F.J. Abdullayeva, “Advanced Cloud computing persistent threat assault detection technique based on softmax regression and autoencoder algorithm,” Array, vol. 10, pp. 1-11.

[8] I. Guyon, A. Elisseeff, “An introduction to variable and feature selection,” Journal of Machine Learning Research, 2003, vol. 3, pp. 1157-1182.

[9] R. Battiti, “IEEE Transactions on Neural Networks, 1994, Vol. 5, no. 4, pp. Using mutual information to select features in supervised neural net learning 537-550”.r https://doi.org/10.1109/72.298224.

[10] CSE-CIC-IDS2018 on AWS, https://www.unb.ca/cic/datasets/ids- 2018.html

[11] HTTP DATASET CSIC 2010, Information Security Institute

[12] Alaigwu, B. E., & Chiroma, H. (2021). A systematic review of cloud computing security: Issues and challenges. Journal of Network and Computer Applications, 176, 102918.

[13] Hameed, S., & Khan, F. (2022). A hybrid feature selection approach for intrusion detection in cloud environments. Future Generation Computer Systems, 127, 345–357.

[14] Moustafa, N., Creech, G., Slay, J., & Turnbull, B. (2020). Big data analytics for intrusion detection: Advances, challenges, and opportunities. Future Generation Computer Systems, 95, 476 -489.

[15] Singh, S., Chana, I., & Buyya, R. (2020). Agile cloud security service composition for proactive intrusion detection. IEEE Transactions on Cloud Computing, 9(2), 637–650.

[16] Zhang, H., & Wu, S. (2021). Zero-day attack detection in cloud environments using ensemble learning techniques. IEEE Access, 9, 78945–78955.

[17] Shafiq, M. O., Yu, X., Bashir, A. K., & Chaudhry, S. A. (2023). AI-driven DDoS detection using hybrid deep learning models in multi-cloud architectures. Computers & Security, 126, 103013.

[18] Alsaedi, N., & Moustafa, N. (2021). A novel graph-based feature selection technique for cyberattack detection in cloud networks. Journal of Information Security and Applications, 57, 102703.

[19] Sahu, A. K., Yadav, D. K., & Pateriya, R. K. (2020). A feature selection based ensemble classifier for effective intrusion detection. Procedia Computer Science, 167, 2350–2359. https://doi.org/10.1016/j.procs.2020.03.289.