Journal of Science Engineering Technology and Management Science
Volume 02, Issue 08, August 2025 *ISSN: 3049-0952*
www.jsetms.com *DOI: 10.64771/jsetms.2025.v02.i08.541-547*

ML-BASED DETECTION AND PREVENTION OF PRIVILEGE ESCALATION ATTACKS IN CLOUD ENVIRONMENTS

P. Manjulatha[1], Y. Pavan Sai[2], G. Rakshit Kumar[2], K. Chandu Vara Prasad[2], M. Shiva Prasad[2]
[1]Assistant Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering (CSIT)
[1,2]Sree Dattha Institute of Engineering and Science, Sheriguda, Ibrahimpatnam, 501510, Telangana.

To Cite this Article
P. Manjulatha, Y. Pavan Sai, G. Rakshit Kumar, K. Chandu Vara Prasad, M. Shiva Prasad, "ML-Based Detection and Prevention of Privilege Escalation Attacks in Cloud Environments", Journal of Science Engineering Technology and Management Science, Vol. 02, Issue 08, August 2025,pp: 541-547, DOI: http://doi.org/10.64771/jsetms.2025.v02.i08.pp541-547
Submitted: 15-07-2025 *Accepted: 21-08-2025* *Published: 28-08-2025*

ABSTRACT
Cloud computing has transformed how businesses and individuals store and access data, but it has also introduced critical vulnerabilities—particularly insider threats. These threats arise when employees or privileged users misuse their access to sensitive information. According to the 2022 Cloud Security Report, insider attacks account for approximately 35% of all global cloud data breaches. Detecting such threats in cloud environments is vital to maintaining data integrity, confidentiality, and business continuity.Traditional detection methods—such as rule-based systems, manual audits, and access log analyses—are reactive, time-consuming, and prone to human error. These conventional systems are often ineffective in large-scale cloud infrastructures, where the volume of data is immense and rapidly changing, making timely threat detection difficult.Given the rising frequency of insider threat incidents and the limitations of existing methods, there is a growing need for more advanced and automated solutions. Machine learning, especially ensemble learning models, provides a promising approach to enhancing detection capabilities. By leveraging algorithms like Random Forest, AdaBoost, and CatBoost, these models can efficiently analyze user behavior patterns, identify anomalies, and detect potential threats in real-time. Unlike traditional systems, machine learning models can process vast datasets from cloud logs and user activities more accurately, reducing false positives and significantly improving the effectiveness of security responses.

Keywords: Cloud security, insider threats, ensemble learning, machine learning, anomaly detection, cloud logs, privileged access, real-time detection.

1. INTRODUCTION
Cloud computing has revolutionized organizational data management by offering scalable, flexible, and cost-effective infrastructure. However, this increased reliance on cloud platforms has introduced significant security challenges, particularly insider threats. These threats involve authorized users, such as employees or contractors, misusing their access to compromise or steal sensitive data. In 2022, insider threats accounted for 35% of global cloud breaches, with Indian organizations witnessing a notable rise in such incidents. A 2021 Nasscom report highlighted that 33% of Indian firms experienced cloud security issues, with insider threats being a key contributor.

Fig. 1: Privilege escalation attack detection.

As India accelerates its digital transformation, ensuring data confidentiality, integrity, and availability in cloud systems is more critical than ever. Traditional detection methods—such as manual audits, rule-based systems, and access log reviews—are reactive, error-prone, and ineffective in large-scale environments with dynamic data flows. This growing inadequacy underscores the need for intelligent, automated detection systems. Motivated by the urgent need to secure cloud infrastructures, this research explores the application of ensemble machine learning models to enhance the detection of insider threats. These models—by combining algorithms like Random Forest, AdaBoost, and CatBoost—offer improved pattern recognition, anomaly detection, and real-time threat alerts. The proposed solution aims to provide a scalable and proactive security framework, particularly beneficial to sectors like finance, healthcare, government, and e-commerce, where the sensitivity of cloud-hosted data demands robust defense mechanisms. Additionally, it supports regulatory compliance with standards such as GDPR and HIPAA, ultimately offering a comprehensive approach to insider threat mitigation in modern cloud ecosystems.

2. LITERATURE SURVEY

Insider threat detection is a broadly researched topic; a variety of solutions have been proposed: specifically, different learning techniques to facilitate early, more accurate threat detection. To discover present research gaps and potential future research domains, an analytical review of the various approaches to insider threat identification is required.[1]. Cloud computing prevents people from spending a lot on equipment maintenance and purchases by utilizing cloud infrastructure. Cloud storage providers adopt fundamental security measures for their systems and the data they handle, including encryption, access control, and authentication. Depending on the accessibility, speed, and frequency of data access, the cloud has an almost infinite capacity for storing any type of data in different cloud data storage structures. Sensitive data breaches might occur due to the volume of data that moves between businesses and cloud service providers, both inadvertent and malicious. The characteristics that make online services easy to use for workers and IT systems also make it harder for businesses to prevent unwanted access [2]. Authentication and open Interfaces are new security vulnerabilities that Cloud services subject enterprises face. Hackers with advanced skills utilize their knowl- edge to access Cloud systems Machine learning employs a variety of approaches and algorithms to address the security challenge and better manage data. Many datasets are private and cannot be released owing to privacy concerns, or they may be missing crucial statistical properties [3]. [4]. The fast rise of the Cloud industry creates privacy and security risks governed by regulations. Le et al. [5] discussed that insider threats are among the most expensive and difficult-to-detect forms of assault since insiders have access to a company's networked systems and are familiar with its structure and security processes. A unique set of challenges face insider malware detection, such as extremely unbalanced data, limited ground truth, and behavioral drifts and shifts. Machine learning is used to analyze data at several levels of detail under realistic situations to identify harmful behaviors, especially malicious insider attacks. Random Forest beats the other ML methods, achieving good

detection performance and F1-score with low false positive rates in most situations. The proposed work achieved an accuracy of 85% and a false positive rate of only 0.78%. Janjua et al. [6]
Discussed that preventing mali- cious insiders from acting maliciously in an organization's system is a significant cybersecurity challenge. The paper's main goal is to use several Machine Learning approaches to classify email from the TWOS dataset. The following supervised learning techniques that have been used on the dataset are Adaboost, Naïve Bayes (NB), Logistic Regression (LR), KNN, Linear Regression (LR), and Support Vector Machine (SVM). Experiments reveal that AdaBoost has the best classification accuracy for harmful and non-malicious emails, with a 98% accuracy rate. Although the model was trained on the original dataset, the data is limited. The model's results may be improved if the dataset is bigger. Kumar et al. [7] discussed that due to the large number of diverse apps operating on shared resources, implementing security and resilience on a Cloud platform is necessary but difficult. Inside the Cloud infrastructure. Based on the idea of clustering, a novel malware detection technique was suggested: trend micro locality sensitive hashing (TLSH). They utilized Cuckoo sandbox, which generates dynamic file analysis results by running them in a separate environment. Principal component analysis (PCA), random forest, and Chi- square feature selection approaches are also used to choose the essential features. pathy et al. [8] discussed that conventional web-based and cloud apps are vulnerable to the most popular online threats. One of the greatest threats to a SaaS application is the SQL injection attack. They construct and test the classification for SQL attack detection using machine learning methods. They explore the ability of machine learning models to identify SQL injection attacks, including the AdaBoost Classifier, Random Forest, and Deep Learning utilizing ANN, Tensor- Flow's Linear Classifier, and Boosted Trees Classifier. More important than malicious reading activities are malicious writing operations. The random forest classifier surpasses all others on the dataset and obtains better accuracy. Sun et al. [9] discussed that the network is becoming increasingly integral to businesses and organizations. So there is an increase in network security threats. Data leakage inci- dents from 15 nations and 17 industry groups were examined for Ponemon's 2018 Cost of a Data Breach Study, with 48% being malicious operations. While insiders' faulty actions were the cause of 27% of the incidents. Kim et al.
[10] discussed that the authorized user's malicious acts, such as stealing intellectual property or sensitive information, fraud, and sabotage, are examples of insider risks. Although insider threats are far less common than external network assaults, they can still do significant harm. There are three widely used research methodologies for detecting insider threats. Liu et al.

3. PROPOSED METHODOLOGY

The primary objective of this research is to detect and mitigate insider threats in cloud environments using advanced machine learning techniques, specifically ensemble learning models. Insider threats, carried out by trusted individuals with legitimate access, are challenging to detect; hence, this study focuses on combining classifiers such as Random Forest, AdaBoost, and CatBoost to improve detection accuracy. A user-friendly graphical user interface (GUI) developed with Tkinter enables users to upload datasets, preprocess data, train and evaluate models, and make predictions on new data. The dataset used includes labeled examples of normal and insider threat behaviors, which are loaded through a file dialog, displayed in the interface, and visualized with class distribution graphs. Data preprocessing includes handling missing values, separating features and target variables, standardizing the data, and splitting it into training and testing sets. Each model is trained on the preprocessed data, and performance is assessed using metrics like accuracy, precision, recall, F1-score, and confusion matrices. Among the classifiers, CatBoost is proposed as the most effective due to its advanced boosting technique and superior handling of categorical features. The GUI allows for real-time prediction of insider threats using new test data and displays results instantly. Additionally, a performance visualization feature compares all three models using bar graphs, helping users identify the most accurate and reliable model for deployment in real-world cloud environments.

The CatBoost Classifier is a high-performance, gradient boosting-based ensemble algorithm developed by Yandex. It is particularly efficient for handling categorical features and offers robust performance with minimal tuning. In the context of insider attack detection in cloud environments, the Ensemble CatBoost Classifier offers improved accuracy, faster training, and better generalization over traditional classifiers by automatically handling data types and reducing overfitting.

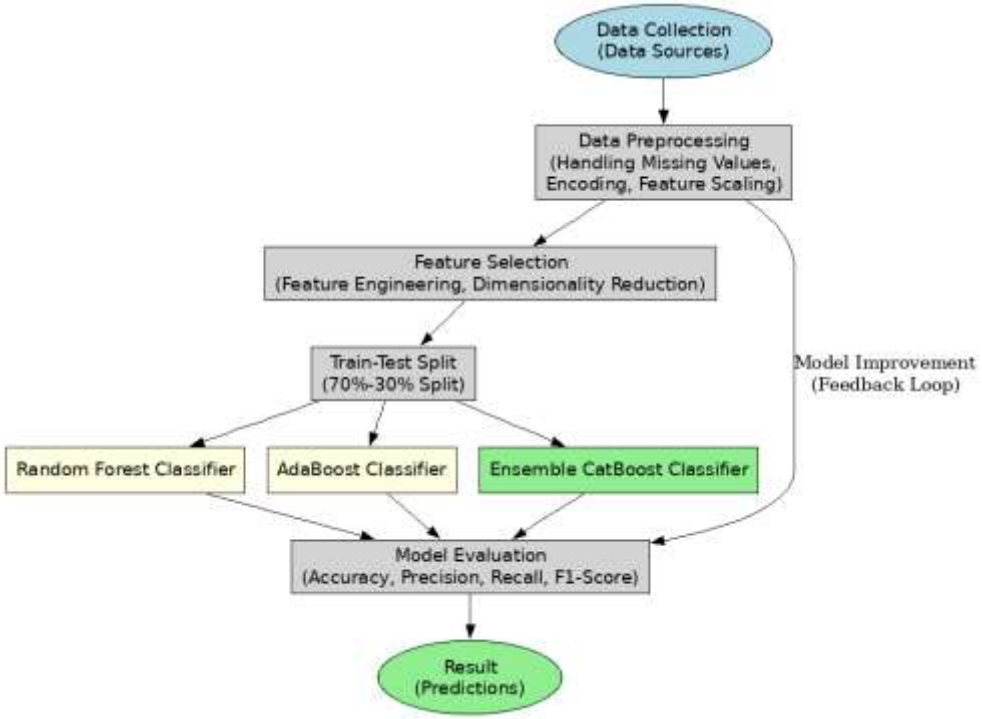

Fig. 2: Block Diagram of Proposed System.

The insider threat detection model is built using structured cloud logs and activity data, requiring minimal preprocessing due to CatBoost's native support for categorical features such as user roles and resource types. The training data (X_train) includes features like user ID, resource accessed, access attempts, timestamps, IP ranges, access frequency, file operations, and behavioral patterns, while the labels (y_train) indicate normal or insider threat activity. CatBoost is trained using ordered boosting to prevent overfitting and target leakage, while also offering symmetric tree construction for fast inference and built-in class balancing for handling imbalanced datasets. When tested on unseen cloud activity logs (X_test), the model delivers high-speed predictions with strong performance on noisy and overlapping data. Evaluation is conducted using ground truth labels (y_test) and metrics such as accuracy, precision, recall, F1-score, confusion matrix, and ROC-AUC, all of which indicate the model's high reliability and threat detection capability.

The ensemble CatBoost classifier offers several advantages including superior accuracy, seamless handling of categorical and numerical features, fast inference time, minimal need for preprocessing, robustness against overfitting, and explainable SHAP-based feature importance, making it highly suitable for real-time, secure cloud environments.

4. RESULTS

The implementation of this research focuses on detecting insider threats in cloud environments through ensemble learning, combining models like AdaBoost and the proposed CatBoost classifier to enhance detection accuracy. The process begins by importing essential Python libraries such as pandas, NumPy, scikit-learn, CatBoost, and visualization tools, followed by loading a structured dataset containing cloud activity logs. This dataset includes detailed user behavior features like timestamps, roles, login frequency, file access patterns, and USB usage, all stored in a DataFrame for analysis. Exploratory Data Analysis (EDA) is conducted to understand feature distribution and detect

imbalances or anomalies. The data is then preprocessed—handling missing values, encoding categorical variables, standardizing numerical values—and split into training and testing sets. Feature selection techniques are applied to reduce dimensionality, ensuring optimal model performance. AdaBoost is trained as the existing model, followed by training CatBoost with parameters optimized for handling both categorical and numerical data. Predictions from both models are generated and evaluated using performance metrics like accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrices. The results are interpreted through visualizations, confirming CatBoost's superior accuracy and robustness in detecting insider threats. The dataset used encompasses a wide range of user activity metrics such as login behaviors, file operations across different file types (e.g., documents, executables), USB interactions, work and after-hour actions, and detailed departmental attributes, making it well-suited for training models that can detect subtle, anomalous patterns indicative of insider attacks.

Dataset loaded

	starttime	endtime	user	...	afterhourhttp_hackf_n-pc2	afterhourhttp_hackf_n-pc3	insider
0	1.262405e+09	1.262491e+09	9	...	0	0	0
1	1.262405e+09	1.262491e+09	31	...	0	0	0
2	1.262405e+09	1.262491e+09	35	...	0	0	0
3	1.262405e+09	1.262491e+09	55	...	0	0	0
4	1.262405e+09	1.262491e+09	64	...	0	0	0
..	...	...	...	...	...	...	
489	1.280718e+09	1.280804e+09	610	...	0	0	1
490	1.280804e+09	1.280891e+09	610	...	0	0	1
491	1.280891e+09	1.280977e+09	610	...	0	0	1
492	1.280977e+09	1.281064e+09	610	...	0	0	1
493	1.281064e+09	1.281150e+09	610	...	0	0	1

[494 rows x 830 columns]

Fig. 3: Uploaded the CERT Dataset in the GUI interface of the project.

The figure 3 shows the dataset loaded into the GUI interface of the project, indicating that the dataset is successfully imported and ready for further analysis or model training.

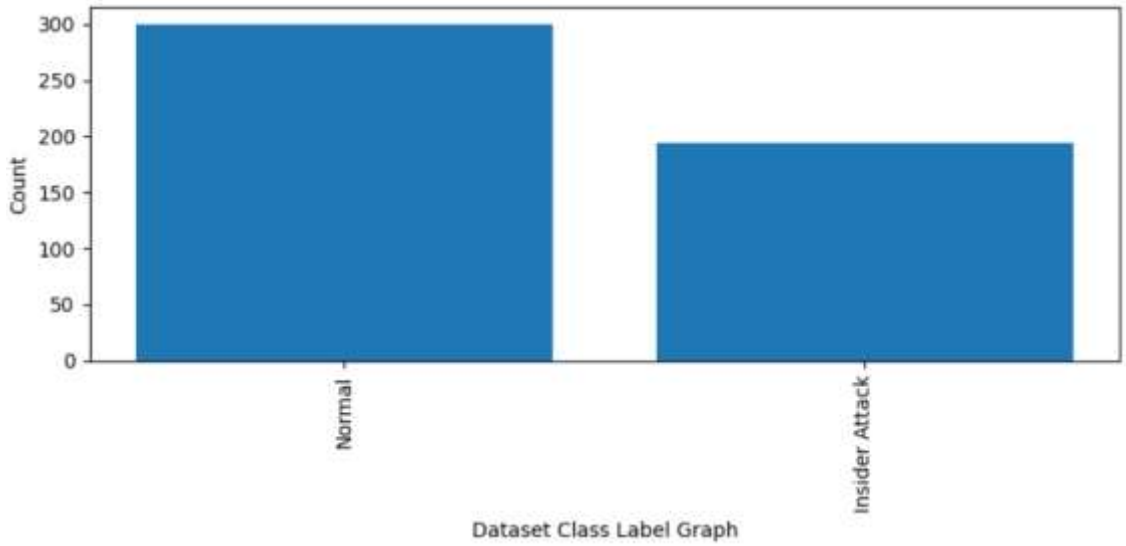

Fig. 4: Count plot for the categories of target variable.

The figure 4 demonstrates the data splitting process, dividing the dataset into training and testing sets. The count plot provides a visual representation of the distribution of target labels, helping in understanding class balance before training the models.

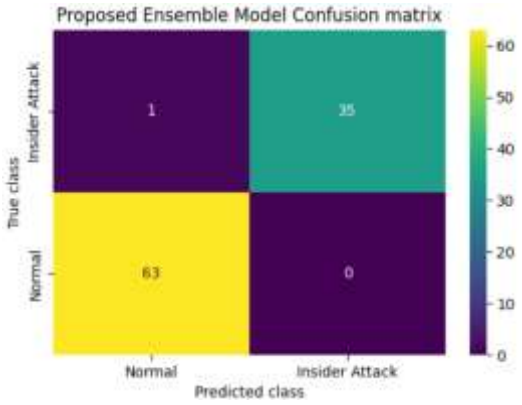

Fig. 5: Confusion Matrix of Existing Proposed Ensembled CatBoost Model.

The figure 5 displays the confusion matrix for the the proposed Ensemble Catboost model, showing their classification performance by comparing true positive, true negative, false positive, and false negative counts.

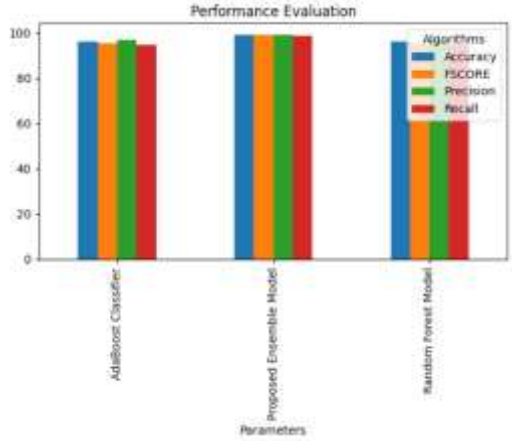

Fig. 6: Performance evaluation graph for comparison.

The figure 6 presents a graph comparing the performance of the RFC, ABC, and the proposed Ensemble Catboost model based on key metrics such as accuracy, precision, recall, and F-Score, illustrating the improvements made by the ensemble model.

```
Test Data = [1.2795948e+09 1.2796812e+09 1.5130000e+03 2.0400000e+02 2.9000000e+01
 1.0000000e+00 0.0000000e+00 1.0000000e+01 4.7000000e+01 1.0000000e+00
 1.0000000e+00 1.6000000e+01 4.1000000e+01 0.0000000e+00 4.0000000e+01
 3.0000000e+01 3.9000000e+01 4.0000000e+01 2.6000000e+01 1.2900000e+02
 1.2900000e+02 0.0000000e+00 0.0000000e+00 0.0000000e+00 1.1300000e+02
 1.1300000e+02 0.0000000e+00 0.0000000e+00 0.0000000e+00 1.6000000e+01
 1.6000000e+01 0.0000000e+00 0.0000000e+00 0.0000000e+00 1.0000000e+00
 1.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 1.0000000e+00
 1.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00] Predicted AS ======> Insider Attack

Test Data = [1.2624048e+09 1.2624912e+09 1.4220000e+03 5.0000000e+00 0.0000000e+00
 0.0000000e+00 1.0000000e+00 5.6000000e+01 2.6000000e+01 1.0000000e+00
 1.0000000e+00 1.0000000e+01 3.1000000e+01 0.0000000e+00 3.3000000e+01
 4.1000000e+01 2.0000000e+01 3.7000000e+01 3.0000000e+01 1.0000000e+00
 1.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 1.0000000e+00
 1.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00
 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00 0.0000000e+00] Predicted AS ======> Normal
```

Fig. 7: Proposed Model Predication on Test Cases.

The figure 7 shows the predictions made by the proposed Ensemble Catboost model on the test data, highlighting how well the model performs on unseen data compared to other classifiers.

5. CONCLUSION

The research concludes that ensemble learning models provide a powerful, automated approach for detecting insider threats in cloud environments by leveraging the combined strengths of algorithms like Random Forest, AdaBoost, and CatBoost. This methodology significantly improves detection accuracy, reduces false positives, and enables real-time threat identification, addressing the inefficiencies and limitations of traditional manual and rule-based systems. Looking ahead, the system can be further enhanced by integrating deep learning models such as LSTM and Transformers for better sequence analysis of user behavior, and by developing real-time analytics dashboards for actionable insights. Adaptive learning will help the model evolve with emerging threat patterns, while industry-specific customization can improve performance in sectors like healthcare and finance. Privacy-preserving techniques like federated learning can support secure, collaborative training, and incorporating automated incident response and dynamic access control can strengthen threat mitigation. The framework can also be scaled for multi-cloud deployments (AWS, Azure, GCP), enriched with global threat intelligence, synthetic data, and aligned with cybersecurity standards like GDPR and ISO 27001 to support compliance and forensic auditing, making it a comprehensive and future-ready solution.

REFERENCES

[1]. U. A. Butt, R. Amin, H. Aldabbas, S. Mohan, B. Alouffi, and A. Ahmadian, ‘‘Cloud-based email phishing attack using machine and deep learning algorithm,’’ Complex Intell. Syst., pp. 1–28, Jun. 2022.

[2]. D. C. Le and A. N. Zincir-Heywood, ‘‘Machine learning based insider threat modelling and detection,’’ in Proc. IFIP/IEEE Symp. Integr. Netw. Service Manag. (IM), Apr. 2019, pp. 1–6.

[3]. P. Oberoi, ‘‘Survey of various security attacks in clouds based environments,’’ Int. J. Adv. Res. Comput. Sci., vol. 8, no. 9, pp. 405–410, Sep. 2017.

[4]. A. Ajmal, S. Ibrar, and R. Amin, ‘‘Cloud computing platform: Performance analysis of prominent cryptographic algorithms,’’ Concurrency Comput.,Pract. Exper., vol. 34, no. 15, p. e6938, Jul. 2022.

[5]. U. A. Butt, R. Amin, M. Mehmood, H. Aldabbas, M. T. Alharbi, and N. Albaqami, ‘‘Cloud security threats and solutions: A survey,’’ Wireless Pers. Commun., vol. 128, no. 1, pp. 387–413, Jan. 2023.

[6]. H. Touqeer, S. Zaman, R. Amin, M. Hussain, F. Al-Turjman, and M. Bilal, ‘‘Smart home security: Challenges, issues and solutions at different IoT layers,’’ J. Supercomput., vol. 77, no. 12, pp. 14053–14089, Dec. 2021.

[7]. S. Zou, H. Sun, G. Xu, and R. Quan, ‘‘Ensemble strategy for insider threat detection from user activity logs,’’ Comput., Mater. Continua, vol. 65, no. 2, pp. 1321–1334, 2020.

[8]. G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, ‘‘On the effectiveness of machine and deep learning for cyber security,’’ in Proc. 10th Int. Conf. Cyber Conflict (CyCon), May 2018, pp. 371–390.

[9]. D. C. Le, N. Zincir-Heywood, and M. I. Heywood, ‘‘Analyzing data granularity levels for insider threat detection using machine learning,’’ IEEE Trans. Netw. Service Manag., vol. 17, no. 1, pp. 30–44, Mar. 2020.

[10]. F. Janjua, A. Masood, H. Abbas, and I. Rashid, ‘‘Handling insider threat through supervised machine learning techniques,’’ Proc. Comput. Sci., vol. 177, pp. 64–71, Jan. 2020.