MACHINE LEARNING BASED DETECTION AND MITIGATION OF PRIVILEGE ESCALATION ATTACKS IN CLOUD

Mrs. Lingareddy Lakshmi Tejaswi[1], Ms. Meesala Supriya[2],

[1]Assistant Professor, Department of Master of Computer Applications (MCA), QIS College of Engineering &Technology (Autonomous), Vengamukkapalem (V), Ongole, Prakasam, AP, India

[2]MCA Student, Department of Master of Computer Applications (MCA), QIS College of Engineering &Technology (Autonomous), Vengamukkapalem (V), Ongole, Prakasam, AP, India

Abstract: This project employs advanced machine learning to fortify cloud security, specifically targeting and mitigating privilege escalation attacks for a more robust defense mechanism. As cloud adoption rises, so does the risk of privilege escalation attacks. This project addresses vulnerabilities in employee access privileges within cloud services to enhance overall security. Leveraging machine learning, the project enables real-time detection and mitigation of privilege escalation attacks. Techniques like LightGBM, Random Forest, Adaboost, and Xgboost contribute to a dynamic defense against evolving threats. Users and businesses experience heightened data security, fostering trust in cloud computing. Cloud service providers and enterprises gain confidence in a secure online environment, benefiting from the project's security enhancements. And also included, a Voting Classifier, amalgamating predictions from Decision Tree, Random Forest, and Support Vector Machine through a "soft" voting approach, enhances the system's performance in detecting and mitigating privilege escalation attacks. Additionally, a user-friendly Flask framework with SQLite integration optimizes user testing, providing secure signup and signin functionalities for practical implementation and assessment.

Index Terms - Privilege escalation, insider attack, machine learning, random forest, adaboost, XGBoost, LightGBM, classification.

1.INTRODUCTION

Cloud computing is a new way of thinking about how to facilitate and provide services through the Internet. The current 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 knowledge 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. Employee access privileges may not necessarily change when they change roles or positions within the Cloud Company. As a result, old privileges are used inconveniently to steal and harm valuable data. Each account that communicates with a computer has some level of authority. Server databases, confidential files, and other services are often restricted to approved users. A malicious attacker can access a sensitive system by gaining control of a higher user account and exploiting or expanding privileges. Based on their objectives, attackers can move horizontally to obtain control of more systems or vertically to obtain admin and root access till they have complete control of the whole environment [1]. When a user gets the access permissions of another user with the same access level, this is known as horizontal privilege escalation. An attacker can use horizontal privilege escalation to access data that does not necessarily

relate to him. An attacker may be able to uncover holes in a Web application that provides him entry to certain other people's information in badly designed apps [3], [5]. Because the attacker has completed a horizontal elevation of privileges exploit, they can see, alter, and copy sensitive information.

Attackers target data sources because they have the most valuable and sensitive information. Every cloud user's privacy and security are affected if data is lost. Insider threats are harmful operations carried out by people with authorization. With the fast growth of networks, many companies and organizations have established their internal networks. According to recent estimates, 90% of businesses believe they are vulnerable to insider assaults [7]. Attackers can use privilege elevation to open up additional attack routes on a target system. Insider attackers try to get higher privileges or access to more sensitive systems by attempting privilege escalation. Insider attacks are difficult to identify and prevent because they exist beneath the enterprise-level security defense measures and frequently have privileged access to the network. Detecting and classifying insider threats has become difficult and time-consuming [8].

In recent studies, researchers worked on detecting and classifying privileged elevation attacks from insider personnel. They proposed different machine learning and deep learning techniques to counter these challenges. Techniques like SVM, Naïve Bayes, CNN, Linear Regression, PCA, Random Forest, and KNN were applied in recent studies. However, the demand for fast and effective machine learning algorithms is highly valued with the diversity of attack types. Therefore an effective and efficient strategy is required to detect, classify and mitigate these insider attacks. To get better security protection systems, we need intelligent algorithms, such as ML algorithms, to classify and predict insider attacks [17].

In addition, knowing the performance of ML algorithms on classifying insider attacks allows you to choose the most appropriate algorithm for each case, and the ones (ML algorithms) need to be improved. So you can provide a higher level of security protection. This research aims to apply effective and efficient ML algorithms to insider attack scenarios to gain better and faster

results. ML algorithms have been applied and evaluated in this regard: Random Forest, AdaBoost, XGBoost, and LightGBM. The principle behind the boosting strategy is to take a weak classifier and train it to become a very good one by raising the prediction of the classification algorithm. Random Forest, AdaBoost, and XGBoost worked accurately and quickly to classify insider threats.

2.LITERATURE REVIEW

Cloud computing is the on-demand availability of PC framework resources. Especially information storage and handling power, without direct unique administration by the customer. It has provided customers with public and private computing and data storage on a single platform across the Internet. Aside from that, it faces several security threats and issues, which may slow down the adoption of cloud computing models. [5] Cloud computing security threats, difficulties, strategies, and solutions are discussed in this paper. Numerous people raised security concerns in a previous survey. Another survey looks at the cloud computing architectural model, and a few of them detail security challenges and techniques. This article brings together all the security concerns, difficulties, techniques, and solutions in one place.

Cloud computing refers to the on-demand availability of personal computer system assets, specifically data storage and processing power, without the client's input. Emails are commonly used to send and receive data for individuals or groups. Financial data, credit reports, and other sensitive data are often sent via the Internet. [1] Phishing is a fraudster's technique used to get sensitive data from users by seeming to come from trusted sources. The sender can persuade you to give secret data by misdirecting in a phished email. The main problem is email phishing attacks while sending and receiving the email. The attacker sends spam data using email and receives your data when you open and read the email. In recent years, it has been a big problem for everyone. This paper uses different legitimate and phishing data sizes, detects new emails, and uses different features and algorithms for classification. A modified dataset is created after measuring the existing approaches. We created a feature extracted comma-separated values (CSV) file and label file, applied the support vector machine (SVM) [8, 10], Naive Bayes (NB), and long short-term

memory (LSTM) algorithm [1, 27]. This experimentation considers the recognition of a phished email as a classification issue. According to the comparison and implementation, SVM, NB and LSTM performance is better and more accurate to detect email phishing attacks. The classification of email attacks using SVM, NB, and LSTM classifiers achieve the highest accuracy of 99.62%, 97% and 98%, respectively.

With advancements in science and technology, cloud computing is the next big thing in the industry. Cloud cryptography is a technique that uses encryption algorithms to secure data [4]. The significant advantage of cloud storage is no difficulty to get to, diminished equipment, low protection, and fixing cost so every association is working with the cloud. Encryption is the process of encoding information to prevent unauthorized access. Nowadays, we desire to secure the information that is to be stored in our computer or transmitted utilizing the internet against attacks. [4] The cryptographic method depends on their response time, confidentiality, bandwidth, and integrity. Furthermore, security is a significant factor in cloud computing for ensuring client data is placed on the safe mode in the cloud. Our research paper compares the efficiency, usage, and utility of available cryptography algorithms. Evaluation results suggest which algorithm is better for which type of data and environment.

With the wide use of technologies nowadays, various security issues have emerged. Public and private sectors are both spending a large portion of their budget to protect the confidentiality, integrity, and availability of their data from possible attacks. Among these attacks are insider attacks which are more serious than external attacks, as insiders are authorized users who have legitimate access to sensitive assets of an organization [36]. As a result, several studies exist in the literature aimed to develop techniques and tools to detect and prevent various types of insider threats. This article reviews different techniques and countermeasures that are proposed to prevent insider attacks. A unified classification model is proposed to classify the insider threat prevention approaches into two categories (biometric-based and asset-based metric). [36, 37] The biometric-based category is also classified into (physiological, behavioral and

physical), while the asset metric-based category is also classified into (host, network and combined). This classification systematizes the reviewed approaches that are validated with empirical results utilizing the grounded theory method for rigorous literature review. Additionally, the article compares and discusses significant theoretical and empirical factors that play a key role in the effectiveness of insider threat prevention approaches (e.g., datasets, feature domains, classification algorithms, evaluation metrics, real-world simulation, stability and scalability, etc.). Major challenges are also highlighted which need to be considered when deploying real-world insider threat prevention systems. Some research gaps and recommendations are also presented for future research directions.

The Internet of Things [34] is a rapidly evolving technology in which interconnected computing devices and sensors share data over the network to decipher different problems and deliver new services. For example, IoT is the key enabling technology for smart homes. Smart home technology provides many facilities to users like temperature monitoring, smoke detection, automatic light control, smart locks, etc. However, it also opens the door to new set of security and privacy issues, for example, the private data of users can be accessed by taking control over surveillance devices or activating false fire alarms, etc. These challenges make smart homes feeble to various types of security attacks and people are reluctant to adopt this technology due to the security issues. In this survey paper [6], we throw light on IoT, how IoT is growing, objects and their specifications, the layered structure of the IoT environment, and various security challenges for each layer that occur in the smart home. This paper not only presents the challenges and issues that emerge in IoT-based smart homes but also presents some solutions that would help to overcome these security challenges.

3.METHODOLOGY

i) Proposed Work:

The proposed system is a machine learning-based solution for insider threat detection and classification in cloud environments. Utilizing Random Forest, Adaboost, XGBoost, and LightGBM algorithms enhances prediction performance. The proposed system achieves improved accuracy in detecting insider threats by leveraging

multiple machine learning algorithms, including Random Forest, Adaboost, XGBoost [35], and LightGBM. Utilizing ensemble learning techniques, the system combines the strengths of various algorithms, enhancing the overall predictive performance for insider threat detection in cloud environments. The system employs robust data preprocessing techniques, including data aggregation and normalization, addressing challenges such as missing values, outliers, and irrelevant features for better model performance. Parameters such as learning rate, maximum depth, and K-fold are tuned to optimize the efficiency of the machine learning models, ensuring a more effective and tailored approach to insider threat detection. And also included a Voting Classifier, amalgamating predictions from Decision Tree, Random Forest, and Support Vector Machine [10] through a "soft" voting approach, enhances the system's performance in detecting and mitigating privilege escalation attacks. Additionally, a user-friendly Flask framework with SQLite integration optimizes user testing, providing secure signup and signin functionalities for practical implementation and assessment.

ii) System Architecture:

The system architecture comprises four key stages: data collection, data preprocessing, application of supervised machine learning algorithms, and results analysis. In the data collection phase, a customized dataset derived from multiple files of the CERT dataset is utilized. Subsequently, the collected data undergoes preprocessing, involving techniques such as data aggregation, normalization, and feature extraction to enhance its quality and relevance. The core of the system involves applying machine learning algorithms—Random Forest, AdaBoost, XGBoost, and LightGBM [31, 32] and a voting classifier as an extension—to the preprocessed data for the detection and classification of privilege escalation attacks. Finally, the system conducts a thorough analysis of the results, evaluating the performance of each algorithm and providing insights into the effectiveness of the overall system in identifying insider threats. This architecture ensures a systematic and robust approach to addressing privilege

escalation attacks through machine learning techniques.

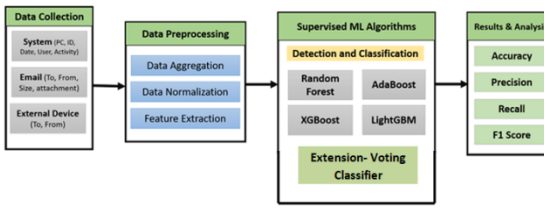

Fig 1. Proposed architecture

iii) Dataset collection:

The dataset employed in this project is derived from multiple files within the CERT dataset [13, 14], specifically focusing on email-related information. This curated dataset captures a variety of instances relevant to insider threat scenarios within email communications. It includes diverse features and attributes related to user behavior, email content, and system interactions.

	id	date	user	pc	to	cc	bcc
0	{R37-B4TX96FG-8219JVIFF}	01/02/2010 07:11:45	LAP0338	PC-5758	Dean.Flynn.Hines@dtaa.com;Wade_Harrison@lockhe...	Nathaniel.Hunter.Heath@dtaa.com	NaN
1	{R8R9-E4GL59IK-2907OSIVU}	01/02/2010 07:12:16	MOH0273	PC-6699	Odonnell-Gage@bellsouth.net	NaN	NaN
2	{I2B2-A8XY59CP-2847ZJZL}	01/02/2010 07:13:00	LAP0338	PC-5758	Penelope_Colon@netzero.com	NaN	NaN
3	{A3A9-F4TH89AA-8318GFGK}	01/02/2010 07:13:17	LAP0338	PC-5758	Judith_Hayden@comcast.net	NaN	NaN
4	{E8B7-C8FZ88UF-2946RUQQ}	01/02/2010 07:13:28	MOH0273	PC-6699	Bond-Raymond@verizon.net;Alea_Ferrell@msn.com...	NaN	Odonnell-Gage@bellsouth.net

Fig 2. CERT dataset

iv) Data Processing:

Data processing involves transforming raw data into valuable information for businesses. Generally, data scientists process data, which includes collecting, organizing, cleaning, verifying, analyzing, and converting it into readable formats such as graphs or documents. Data processing can be done using three methods i.e., manual, mechanical, and electronic. The aim is to increase the value of information and facilitate decision-making. This enables businesses to improve their operations and make timely strategic decisions. Automated data processing solutions, such as computer software programming, play a significant role in this. It can help turn large amounts of data, including big data, into meaningful insights for quality management and decision-making.

v) Feature selection:

Feature selection is the process of isolating the most consistent, non-redundant, and relevant features to use in model construction. Methodically reducing the size of datasets is important as the size and variety of datasets continue to grow. The main goal of feature selection is to improve the performance of a predictive model

and reduce the computational cost of modeling.

Feature selection, one of the main components of feature engineering, is the process of selecting the most important features to input in machine learning algorithms. Feature selection techniques are employed to reduce the number of input variables by eliminating redundant or irrelevant features and narrowing down the set of features to those most relevant to the machine learning model. The main benefits of performing feature selection in advance, rather than letting the machine learning model figure out which features are most important.

vi) Algorithms:

LightGBM: LightGBM is a gradient boosting ensemble method that is used by the Train Using AutoML tool and is based on decision trees. As with other decision tree-based methods, LightGBM can be used for both classification and regression. LightGBM is optimized for high performance with distributed systems [31, 32].

LightGBM

```
from lightgbm import LGBMClassifier

# Define the hyperparameters as a dictionary
params = {
    'objective': 'binary',   # The objective for binary classification
    'metric': 'auc',   # Metric to optimize during training
    'num_leaves': 40,
    'learning_rate': 0.004,
    'bagging_fraction': 0.6,
    'feature_fraction': 0.6,
    'bagging_frequency': 6,
    'bagging_seed': 42,
    'verbosity': -1,
    'seed': 42,

}

# Create the LGBMClassifier with the specified hyperparameters
lgbm = LGBMClassifier(**params)

lgbm.fit(X_train, y_train)
```

Fig 3. LightGBM

XGBoost: How XGBoost Works - Amazon SageMaker XGBoost is a popular and efficient open-source implementation of the gradient boosted trees algorithm. Gradient boosting is a supervised learning algorithm, which attempts to accurately predict a target variable by combining the estimates of a set of simpler, weaker models [35].

Xgboost

```
import xgboost as xgb

# Create the XGBoost classifier with the specified hyperparameters
xgb_classifier = xgb.XGBClassifier(
    learning_rate=0.1,
    n_estimators=20,
    max_depth=3,
    min_child_weight=2,
    gamma=5,
    subsample=0.7,
    colsample_bytree=0.5,
    objective='binary:logistic',  # For binary classification
    nthread=2,
    scale_pos_weight=2,
    seed=20,
    reg_alpha=3,
    num_parallel_tree=3,
    max_cat_to_onehot=2
)

xgb_classifier.fit(X_train, y_train)
```

Fig 4. XGBoost

AdaBoost: AdaBoost, also called Adaptive Boosting, is a technique in Machine Learning used as an Ensemble Method. The most common estimator

used with AdaBoost is decision trees with one level which means Decision trees with only 1 split. These trees are also called Decision Stumps.

Adaboost

```
: from sklearn.ensemble import AdaBoostClassifier

adaboost_classifier = AdaBoostClassifier(
    n_estimators=10,
    learning_rate=1.0,
    random_state=0
)

adaboost_classifier.fit(X_train, y_train)
```

Fig 5. Adaboost

RF: Random forest is a commonly-used machine learning algorithm trademarked by Leo Breiman and Adele Cutler, which combines the output of multiple decision trees to reach a single result. Its ease of use and flexibility have fueled its adoption, as it handles both classification and regression problems [34].

Random Forest

```
from sklearn.ensemble import RandomForestClassifier

random_forest_classifier = RandomForestClassifier(
    n_estimators=100,
    random_state=0
)


random_forest_classifier.fit(X_train, y_train)
```

Fig 6. Random forest

VC: A Voting Classifier is a machine learning model that trains on an ensemble of numerous models and predicts an output (class) based on their highest probability of chosen class as the output.

Voting Classifier

```
from sklearn.ensemble import VotingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC


# Create individual classifiers
decision_tree = DecisionTreeClassifier(random_state=0)
random_forest = RandomForestClassifier(n_estimators=100, random_state=0)
svm = SVC(probability=True, random_state=0)

# Create the Voting Classifier with the specified classifiers
voting_classifier = VotingClassifier(
    estimators=[('decision_tree', decision_tree), ('random_forest', random_forest), ('svm', svm)],
    voting='soft'  # 'soft' for using class probabilities for voting
)

# Fit the Voting Classifier to the training data
voting_classifier.fit(X_train, y_train)
```

Fig 7. Voting classifier

4.EXPERIMENTAL RESULTS

	Accuracy	Recall	Precision	F1
LightGBM	94.75	50	47.375	48.65212
Xgboost	94.75	50	47.375	48.65212
AdaBoost	95.45	58.01608	90.27778	62.42581
RandomForest	95.45	58.01608	90.27778	62.42581
Voting Classifier	**96.45**	**66.64028**	**96.82903**	**73.90277**

Fig 8. Performance evaluation

So, this is the performance metrics table. And here we can see the algorithm names and the accuracy, precision, recall, fscore, specificity scores secured by them. So, we can see the extension voting classifier has outperformed all other models in all the performance metrics.

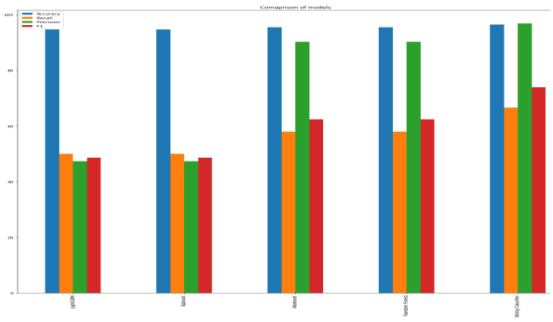

Fig 9. Comparison graph

So, this is the performance metrics comparison graph.

So, here x axis represents algorithm names and y axis represents performance metrics.

So here, blue colour bar represents accuracy, orange denotes recall, green denotes precesion and red is for f1score.

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN}$$

Accuracy: Accuracy is the proportion of correct predictions in a classification task, measuring the overall correctness of a model's predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

F1 Score: The F1 Score is the harmonic mean of precision and recall, offering a balanced measure that considers both false positives and false negatives, making it suitable for imbalanced datasets.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

Fig10. Home Page

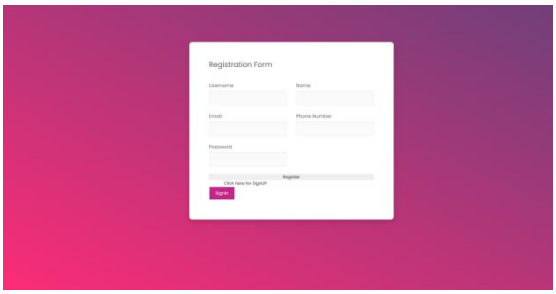

Fig11. Signup Page

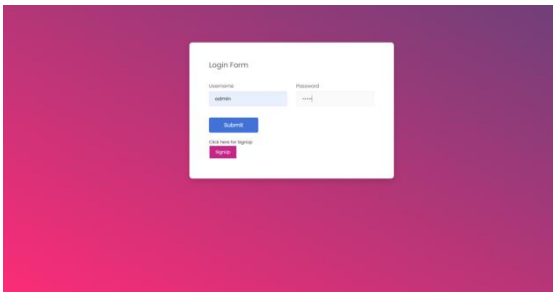

Fig12. Signin Page

Fig 13. User input Page

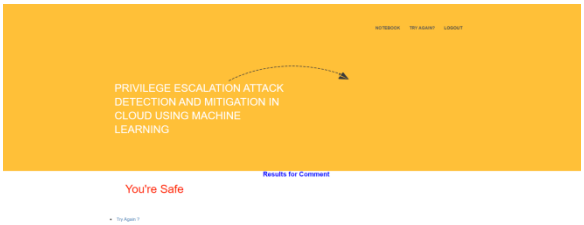

Fig 14. Prediction Result

4.CONCLUSION

The malicious insider becomes a crucial threat to the organization since they have more access and opportunity to produce significant damage. Unlike outsiders, insiders possess privileged and proper access to information and resources. This paper proposed machine learning algorithms for detecting and classifying an insider attack. [14] A customized dataset from multiple files of the CERT dataset is used in this work. Four machine learning algorithms were applied to that dataset and gave better results. These algorithms are Random Forest, AdaBoost, XGBoost, and LightGBM. Using these supervised machine learning algorithms, this paper demonstrated the effective experimental results having higher accuracy in the classification report. Among the proposed algorithms, the LightGBM algorithm provides the highest accuracy of 97%; the other

accuracy values are RF with 86%, AdaBoost with 88%, and XGBoost with 88.27% [31, 32]. In the future, the proposed models may increase their performance by expanding the dataset in size and diversity in terms of its features and the new trends of insider attackers to perform the attack. This may open up new research trends toward detecting and classifying insider attacks related to many fields of organization. Machine learning models are used by businesses to make credible business decisions, and improved model results lead to better judgments. The cost of mistakes can be quite high, however, this cost is reduced by improving model accuracy. ML-based research enables users to provide massive amounts of data to computer algorithms, which then evaluate, recommend, and decide using the supplied data.

4.FUTURE SCOPE

Future enhancements should focus on optimizing the system's scalability to efficiently handle increased workloads in expansive cloud setups, ensuring smooth processing even as data complexity and volume grow. Future developments should implement dynamic response mechanisms capable of rapidly identifying and countering newly emerging tactics in privilege escalation attacks, providing a proactive defense against evolving insider threats. The integration of techniques that provide understandable explanations for model decisions is essential. This transparency helps security analysts comprehend the factors influencing threat identifications, fostering trust in the system's outputs [29, 30]. Establishing a framework for continuously updating and diversifying the dataset used for training the models is crucial. Ongoing enrichment ensures the system's effectiveness in identifying and mitigating new types of attacks and evolving insider threat patterns.

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.

Authors Profile:

Mrs. Lingareddy Lakshmi Tejaswi, currently working as an Assistant professor in the Department of Computer science and Engineering, QIS college of Engineering and Technology, Ongole, Andhra Pradesh. she did her BTech from Rao & Naidu Engineering college, MTech from QISCET. Her area of interest in Machine Learning, Artificial intelligence, cloud computing and programming languages.

Ms. Meesala Supriya, currently pursuing Master of Computer Applications at QIS College of Engineering and Technology (Autonomous), Ongole, Andhra Pradesh. She Completed B.Sc. in Computer Science from Bharathi Degree College, Chirala, Andhra

Pradesh. Her areas of interests are Deep Learning & Machine learning.