Enhancing Insider Threat Detection through Integrated Behavioral, Signature and Anomaly based Detection Methods

Keerthana Palaparthy
Computer Science and Engineering (Cyber Security)
Institute of Aeronautical Engineering
Dundigal, Hyderabad 21951A6214@iare.ac.in

Y. Manohar Reddy Asst. Professor
Computer Science and Engineering (Cyber Security)
Institute of Aeronautical Engineering
Dundigal, Hyderabad y.manoharreddy@iare.ac.in

Jatoth Victor Paul
Computer Science and Engineering (Cyber Security)
Institute of Aeronautical Engineering
Dundigal, Hyderabad 21951A6260@iare.ac.in

S Raju
Computer Science and Engineering (Cyber Security)
Institute of Aeronautical Engineering
Dundigal, Hyderabad 22955A6204@iare.ac.in

***Abstract*— Insider threats present substantial risks to organizational security, as malicious actors exploit their authorized access to systems, networks, or data to perpetrate harmful activities. These threats encompass various forms, including data theft, sabotage, fraud, or espionage, leading to significant financial losses, reputational damage, or regulatory penalties. Traditional approaches to insider threat detection, such as anomaly-based, signature-based, and behavioral analysis methods, have inherent limitations, including high false positives, reliance on known patterns, and lack of contextual understanding. These approaches often fail to classify insider threats accurately, potentially leading to innocent insiders being mislabeled as malicious. In this project, a unified insider threat detection system is proposed, integrating anomaly-based, signature- based, and behavioral analysis methods using Support Vector Machines (SVMs). By combining these methods and leveraging the strengths of SVMs, the aim is to address the limitations of individual approaches and enhance detection accuracy. Weighted voting is employed to fuse the output of each detection method, providing a comprehensive likelihood estimate of insider threats. This integrated approach enables organizations to better identify and mitigate insider threats, safeguarding sensitive assets and maintaining a robust security posture.**

***Keywords*— Insider threat, Insider threat detection, Signature- based detection, Anomaly-based detection, Behavior analysis, False positives, Detection accuracy, Weighted Voting Mechanism.**

I. Introduction

In today's digital age, organizations confront a major threat from insiders— Individuals such as employees, contractors, or business partners who have authorized access to company resources but misuse that access for harmful purposes. Insider threats can cause major financial losses, reputational damage, and the compromise of critical information. Detecting insider threats is challenging because they might be subtle and deceiving.

[25] The purpose of this study is to improve the identification of insider threats using an advanced, integrated system that integrates numerous detection approaches. It uses behavioral analysis, anomaly-based detection and signature-based detection techniques, all powered by Support Vector Machines (SVMs), to accurately and reliably detect possible threats.

Understanding the Magnitude of Insider Threats

- **Prevalence and Impact**: Insider threats have become a significant issue in today's digital landscape. According to the 2023 Insider Threat Report, almost 60% of companies experienced at least one insider attack in the past year. The significant financial consequences are highlighted by the projected average cost of an insider incident, which is around $11.45 million.

- **Growth Trends**: The risk of insider attacks has increased as remote employment and business process digitization have become more common. 34% of all data breaches are caused by insider threats, according to the Verizon 2023 Data Breach Investigations Report, highlighting the critical need for efficient detection and prevention techniques.

Insider Threats - types: 21-Insider threats fall into two categories: malicious insiders, who intentionally harm the organization, and negligent insiders, who unintentionally jeopardize security through careless actions. Both kinds present serious concerns and need for various methods of identification and mitigation.

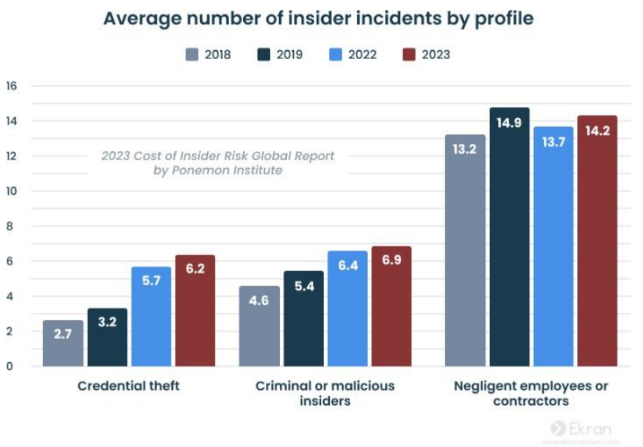


Fig. 1. Average number of insider incidents by profile (Statistics for Insider Threats 2024: Facts, Costs & Reports | Ekran System)

A diversified strategy that incorporates multiple detection methods—each intended to spot hostile activity from a distinct angle—is necessary to counter these threats. In order to find departures from typical behavior that can point to malevolent intent, behavioral detection examines user activity patterns, including login times, access patterns, and data interactions. Organizations can identify tiny changes that could indicate an insider threat by keeping an eye on these patterns. The "signatures" or predetermined patterns of recognized dangers are the foundation of signature-based detection. A user's action is marked as possibly harmful when it meets any of these signatures. Threats that have been previously encountered and documented can be effectively identified using this strategy. Finding odd patterns in data that diverge from predicted behavior is the main goal of anomaly-based detection. By pointing to possible security lapses or malevolent actions that deviate from accepted practices, these anomalies can help identify new and developing risks. Support Vector Machine (SVM), a potent supervised machine learning method, is used in this study to categorize user behavior and identify possible dangers. Because SVMs are resilient against overfitting and successful in high-dimensional spaces, they are especially well-suited for this task. We employ Weighted Voting, a technique that aggregates the outputs of various models or detection methods, to improve the precision and dependability of our detection system. Every approach adds a "vote" to the ultimate choice, with more accurate and dependable approaches receiving more weight. We utilize the Synthetic Minority Over-sampling Technique (SMOTE) to tackle the problem of unbalanced datasets, which is prevalent in insider threat detection. By creating synthetic samples for the minority class, SMOTE gives machine learning models a more balanced dataset to train on, which enhances their performance. We also employ One-Class SVM, which is a variant of SVM designed especially for anomaly identification. One-Class SVM is a useful tool for identifying insider threats that display unusual behavior because it is trained just on the "normal" class and seeks to find outliers or anomalies in the data. This research attempts to create a strong and all-encompassing insider threat detection system that uses behavioral, signature-based, and anomaly-based methodologies to protect enterprises from internal security threats by combining these cutting-edge approaches.

The rest of the paper is organized as follows: Section II provides a summary of relevant literature, and Section III presents the proposed anomaly detection methodology. Section IV outlines the experiments and presents the evaluation results, while Section V delves into a detailed analysis and comparisons. Finally, Section VI provides the conclusion and suggestions for future research.

II. Literature Review

Detecting insider threats within organizations has become a prominent focus in recent years. These threats, [27] posed by individuals within an organization who misuse their access to confidential information, can lead to data breaches and other malicious activities. In this literature review, shows an overview of the key research contributions in the field of insider threat detection, focusing on different methodologies and how effective they are. [1] introduces a novel threat detection scheme that uses beta mixture-hidden Markov models (MHMMs) to identify anomalous activities within industrial systems. Their approach focuses on the dynamic interactions between physical and network systems in Industry 4.0 environments. By modeling these interactions, the proposed scheme can detect both physical and cyber threats with high precision. The evaluation on multiple datasets shows that this method outperforms traditional detection techniques, offering a robust solution for modern applications.

[20] explored the use of unsupervised ensemble learning methods to detect insider threats. Their study emphasizes the importance of combining multiple unsupervised algorithms to improve detection accuracy. By leveraging the strengths of different detection techniques, this approach enhances the identification of anomalous behavior. It's particularly useful in situations where labeled data is scarce, making it valuable for real-world applications. [23] presents a framework for detecting malicious insider threats in cloud environments using supervised learning methods. Their approach leverages machine learning models to analyze user activities and detect anomalies that may indicate insider threats. The study emphasizes the importance of cloud-specific considerations, such as scalability and real time processing, in developing effective detection systems. The proposed framework demonstrates high accuracy in identifying insider threats, making it a promising solution for cloud-based infrastructures. [24] investigates the application of machine learning techniques for detecting insider threats within university website clusters. Their study focuses on developing a detection method that can identify malicious activities by analyzing web traffic patterns and user behavior. To accurately detect potential threats, they use a combination of feature extraction and classification algorithms. This research highlights the unique challenges posed by the academic environment and offers tailored solutions to address these issues. [26] provides a comprehensive review of recent advances in machine learning techniques for detecting malicious insider threats. They discuss the challenges and opportunities associated with various machine learning methods, including supervised, unsupervised, and semi-supervised approaches. The review highlights the need for robust, adaptive systems that can handle the dynamic nature of insider threats. It also evaluates different algorithms and identifies future research directions to enhance detection capabilities.

III. Integrated insider threat Detection System

Figure 2 presents an overview of the proposed insider threat detection system. The system processes raw user activity log data and generates a set of numerical attributes, organized by day or week. Section III-A explains the pre-processing and feature extraction steps in detail. After the characteristics are extracted, supervised learning is utilized to train several detection models. Section III-C provides specifics on how the detection methods were integrated.

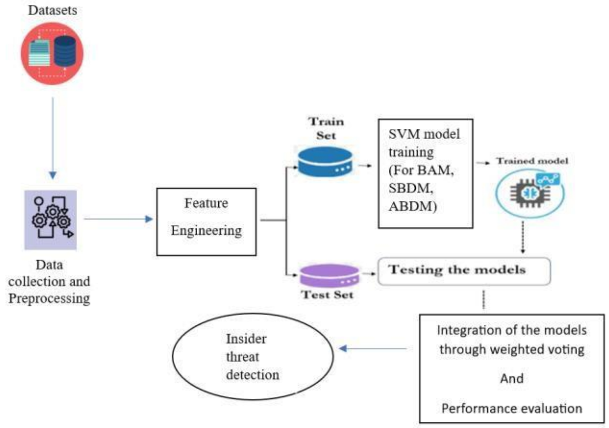


Fig. 2. Proposed system components.

The system's capacity to distinguish between intentional and unintentional threats is a crucial component of this study. Despite the fact that both kinds of threats could appear as anomalies, their root origins and effects are very different:

- **Malicious Threats:** These are intentional actions aimed at causing harm or gaining unauthorized access to resources. Malicious threats often exhibit patterns that significantly deviate from normal behavior, as captured by the anomaly detection model. Signature-based detection further helps identify known malicious patterns, providing a comprehensive analysis.

- **Accidental Threats:** These are unintentional actions that may compromise security, often resulting from user mistakes or negligence. While accidental threats can also appear as anomalies, the behavior-based detection model helps distinguish them by analyzing the context and patterns of user activities. For instance, a sudden spike in activity due to a user's unfamiliarity with a new system can be identified as accidental rather than malicious.

The system allows for adjusting the investigation threshold based on the available resources and the desired sensitivity of detection. The detection methods are demonstrated in Fig. 3. Adjusting the investigation budget (IB)—the percentage of data that security analysts are able to review—allows control over the balance between false positives and false negatives. This adaptability guarantees that the system can be customized to meet various organizational requirements and financial limitations.

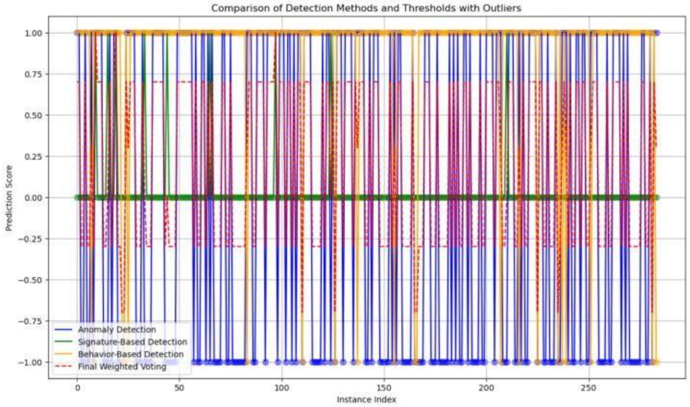


Fig. 3. Demonstration of Detection methods and thresholds with outliers

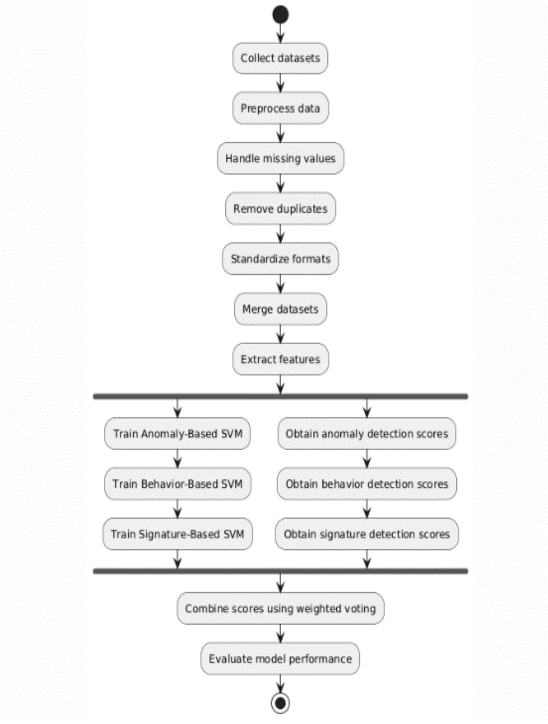

Collect datasets
Preprocess data
Handle missing values
Remove duplicates
Standardize formats
Merge datasets
Extract features
Train Anomaly-Based SVM
Train Behavior-Based SVM
Train Signature-Based SVM
Obtain anomaly detection scores
Obtain behavior detection scores
Obtain signature detection scores
Combine scores using weighted voting
Evaluate model performance

Fig. 4. Process flow of the Proposed Methodology

The work flow of the suggested methodology for creating the integrated insider [26] threat detection system is shown in Fig. 4.

A. Data Pre-Processing and Feature Engineering

Important phases in the research to improve insider threat detection are feature engineering and data pretreatment. These procedures main goal is to ensure quality, consistency, and relevance in order to prepare raw data for efficient modeling.

Data Preprocessing

- **Data Cleaning:**
  Initially, the datasets were examined for missing or inconsistent values. Rows with missing entries were either filled with appropriate statistical measures, such as mean or median, or removed if they were deemed insignificant. This step is essential to maintain the integrity of the analysis and prevent skewed results.

- **Data Integration:**
  To create a comprehensive dataset, various sources were integrated. This included merging user activity data, logon records, decoy file accesses, and psychometric profiles to form a unified view of user behavior. This integration helps in capturing complex interactions and dependencies among different features.

- **Date Conversion:**
  To make time-based analysis easier, date fields were transformed into a datetime format. This makes it possible to perform more complex temporal analysis, such finding trends over predetermined periods of time.
- **Label Encoding:**
  Label encoding or one-hot encoding, as appropriate, was used to convert categorical variables to numerical representations. For algorithms that need numerical input while maintaining the information provided by categorical characteristics, this transformation is essential.

Feature Engineering

- **Activity Counts:**
  Features representing user activity were derived by aggregating counts of interactions from different datasets. For instance, the number of logon events and file accesses were computed for each user to create a profile of normal versus abnormal behavior.
- **Statistical Features:**
  To represent the variety in user activity, statistical measures including mean, standard deviation, and quantiles were computed in addition to simple counts. These features help in distinguishing between regular patterns and outlier behaviors indicative of potential threats.

- **Anomaly Indicators:**
  Based on activity counts, thresholds were established to flag anomalies. Users exceeding the 95th percentile of activity were labeled as suspicious, which helps in identifying potential insider threats effectively.

- **Psychometric Features:**
  Psychometric data were transformed into features that capture user traits and behaviors. These features were integrated into the detection framework to enrich the model's understanding of user profiles, thus aiding in the identification of abnormal activities.

- **Feature Scaling:**
  To ensure all features contribute equally to the modeling process, feature scaling was performed using standardization. This process involved transforming features to have a mean of zero and a standard deviation of one, which is especially important for algorithms sensitive to feature magnitudes, such as Support Vector Machines (SVM).

B. Model Training

In order to create prediction models that can precisely identify possible security dangers based on a variety of input features; model training is an essential step in insider threat detection. To guarantee the stability and efficacy of the models used, this procedure entails a number of crucial procedures.

Selection of Algorithms

In our approach, we utilized several machine learning algorithms tailored for specific detection tasks. For anomaly detection, we opted for One-Class SVM, which is particularly suited for identifying outliers in datasets characterized by a significant imbalance between normal and anomalous instances. For signature-based detection, Support Vector Classifier (SVC) was employed, enabling effective classification of user activities based on established access patterns.

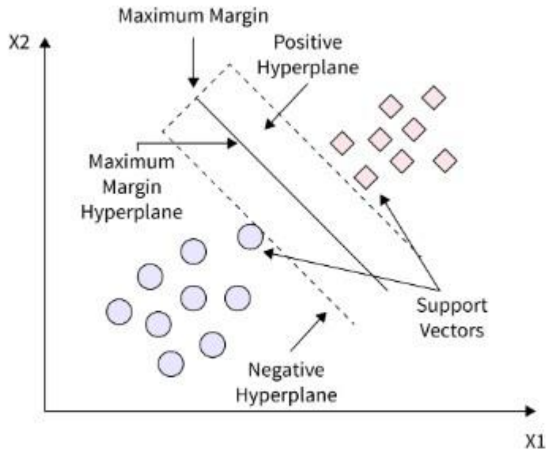


Fig. 5. An example of Support Vector Machines (Non-Linear SVM - Scaler Topics)

Data Preparation

Before training, the datasets underwent extensive preprocessing to enhance the quality of the input features. This included scaling the features using StandardScaler to standardize the data distribution and mitigate the impact of varying feature scales o address class imbalance and ensure the model was trained on a balanced dataset representing both normal and abnormal activities, we used techniques like [24] Synthetic Minority Over- sampling Technique (SMOTE).

Training Process

During the training phase, the processed datasets were split into training and testing subsets. It was standard procedure to set aside 20% of the data for evaluation and 80% for training. After that, the models were fitted to the training data, which enable them [18] to discover underlying correlations and patterns present in the dataset. To determine the optimal configurations for each model, hyperparameters were adjusted using strategies including grid search and randomized search.

Evaluation and Validation

After training, the models' performance was assessed using the designated test set. In order to evaluate the models' ability to detect fraudulent behaviour, crucial parameters such as accuracy, precision, F1-score and recall were calculated. This evaluation process identified potential areas for improvement and provided information about the forecasting ability of the models.

C. Integration of the trained models

One of the most important steps in creating a thorough insider threat detection system is integrating trained models. In order to improve the overall effectiveness and dependability of the threat detection framework, this procedure combines the outputs of different detection algorithms. We can produce predictions that are more reliable and accurate by utilizing the advantages of many models.

Purpose of Model Integration

The primary goal of integrating trained models is to create a synergistic effect that improves the detection capabilities of the system. Individual models may excel in identifying specific types of threats—such as behavioral anomalies, signature-based attacks, or system access irregularities—but may struggle in other areas. By combining these models, we can ensure a more holistic assessment of user activities and system interactions.

Integration Framework

In this research, we utilized an ensemble approach to integrate three distinct types of models: behavior-based detection, anomaly detection, and signature-based detection. Every model offers a distinct perspective, and the sum of their results creates a single prediction framework.

- **Anomaly Detection Model:** Utilizing One-Class SVM, this model identifies outlier behaviors within user activities. It is particularly effective for flagging unusual patterns that deviate from established norms.

- **Signature-Based Detection Model:** Implementing SVC, this model recognizes known patterns of malicious activity based on historical data. It assesses whether specific behaviors align with pre-defined signatures of known threats.

- **Behavior-Based Detection Model:** This model analyzes user behaviors in conjunction with psychological profiles, helping to uncover deviations that may indicate potential insider threats.

Weighted Voting Mechanism

To effectively combine the outputs of these models, we employed a weighted voting mechanism. A particular weight is given to each model's prediction according to its dependability and significance for the detection task as a whole. For instance, the anomaly detection model might carry a higher weight due to its ability to identify previously unseen threats, while the signature model might provide crucial context for known issues.

The final decision regarding user activity is determined by aggregating the predictions from all models. By ensuring that any highlighted behaviour is supported by many sources of information, this method not only increases detection accuracy but also reduces the possibility of false positives.

Performance Evaluation

After integrating the models, we conducted extensive evaluations to assess the effectiveness of the combined framework. Metrics such as F1-score, accuracy, and recall were calculated to assess the overall performance. The results demonstrated that the integrated system outperformed independent models by a significant margin, highlighting the benefits of a collaborative approach to threat identification.

D. *Evaluation*

A crucial component of every machine learning project is evaluation, especially when it comes to insider threat detection, when accuracy is crucial and the stakes are high. The methods and metrics used to evaluate the performance of our integrated detection framework are described in this part, with an emphasis on both the system's overall efficacy and the performance of individual models.

We used a number of crucial indicators to thoroughly assess our detection models' performance:

Precision: Out of all the model's positive predictions, this metric displays the proportion of actual positive forecasts. High precision is necessary to reduce false positives, which can lead to unnecessary alerts and resource allocation.

Recall: Recall is defined as the proportion of genuine positive predictions to the total number of positive events in the dataset. A high recall rate is required to reduce missed detections and ensure that the greatest number of genuine threats are identified.

F1-Score: A balanced metric that accounts for both false positives and false negatives is the F1-score. It is computed as the precision and recall harmonic means. This statistic is particularly useful when there is an imbalance in the distribution of classes because it combines both measures into a single result.

Accuracy: While accuracy provides a general sense of the model's performance, it is particularly helpful when the distribution of classes is reasonably balanced. However, when it comes to anomaly detection, it needs to be evaluated carefully.

IV. Implementation and Results

Involving systematic methodologies to validate the integrated insider threat detection framework's efficacy.This section outlines the steps taken to design, implement, and evaluate the experiments, ensuring robust and reproducible results.

A. *Datasets overview*

Dataset Source: (https://www.kaggle.com/datasets/mrajaxnp/cert-insider-threat-detection-research?)

The success of this research hinged significantly on the diverse and comprehensive datasets employed, each serving a distinct purpose in the detection of insider threats. The primary datasets utilized include decoy file access logs, device activity logs, logon records, psychometric assessments, and user profile information.

- **Decoy File Access Logs:**
  This dataset logs instances of access to decoy files designed to lure potential malicious insiders. It includes fields such as decoy_filename and pc (personal computer identifier). It assists in locating odd file access patterns that might point to malevolent activity.
- **Device Activity Logs:**
  Logs capturing user interactions with devices over time. Includes fields for user, computer, file_tree, activity, date, and id. Used to track device usage patterns and identify anomalies in user behavior.
- **Logon Records:**
  This dataset records user logon activities, providing insights into login behaviors. Includes id, date, user, pc, and activity. It assists in identifying abnormal login attempts or patterns that could suggest a security threat.

- **Psychometric Assessments:**
  Psychological assessments of users, which may provide contextual information about user behavior. Includes

user_id, employee_name, and psychometric scores (the Big Five personality traits are O, C, E, A, and N). This dataset supports behavioral analysis by correlating psychometric traits with potential insider threat behaviors.

- **User Profile Information:**
  Detailed user profiles, including role-based and activity-based information. Typically includes fields like employee_name, user_id, and role. Used to enhance behavioral models and provide context for user activities.

B. *Environmental Setup*

Data Preparation: The first step involved preprocessing the datasets to ensure they were clean, relevant, and suitable for analysis. This involved encoding categorical variables, addressing missing values, and normalizing features. Each dataset was scrutinized to maintain data integrity and consistency across various detection methods.

Model Selection: We selected three distinct detection models based on their suitability for the specific aspects of insider threat detection:

- **Anomaly Detection Model:** Utilizing One-Class SVM, in order to detect anomalous activity suggestive of possible dangers, this model sought to detect departures from known user behavior patterns.

- **Signature-Based Detection Model:** Using SVM for classification, this model focused on recognizing known malicious patterns based on historical access logs and behaviors, effectively flagging suspicious activities.
- **Behavior-Based Detection Model:** This model employed a combination of psychometric and user activity features to detect behavioral anomalies that may signify insider threats.

C. *Experimental Execution*

Training the Models: Each model was trained separately on the relevant features extracted from the preprocessed datasets. To increase the training phase's robustness, a systematic training process was used, incorporating strategies such k-fold cross-validation. This approach helped mitigate overfitting and ensured generalizability across different subsets of data.

Parameter Tuning: The best-performing settings for each model were found through hyperparameter optimization. Model performance was optimized through the rigorous evaluation of various parameter combinations using techniques like grid search and random search.

Integration of Models: Individual models were trained and assessed, and then their outputs were combined using a weighted voting system. By merging the predictions of each model, this integration was created to maximize detection accuracy and minimize false positives. Each model is given a weight (0.5, 0.3, 0.2) according to how important the behavioral analysis model, anomaly detection model, and signature-based detection model are, respectively.

Assessment of Performance

Evaluation Measures: The performance of each model was assessed using a variety of evaluation metrics, such as accuracy, precision, recall, and F1-score. These measurements gave important information about how well the models detected harmful activity and kept the false positive rate low.

- **Accuracy:** Measure of the overall accuracy of the model's predictions.

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

Where,TP stands for True Positives, TN for True Negatives, FP for False Positives and FN for False Negatives

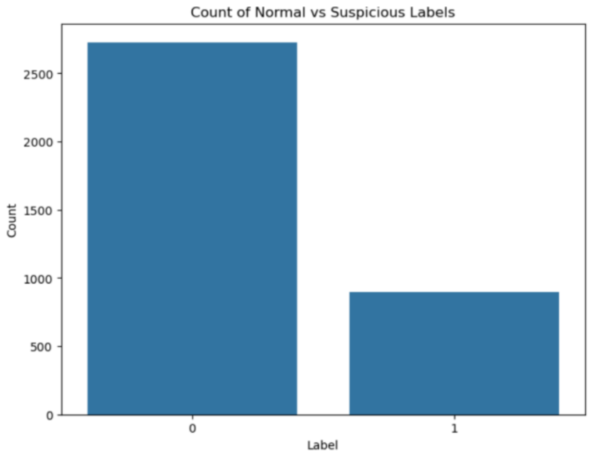
- **Precision:** Percentage of all threats that were accurately predicted to be insider threats.

$$Precision = \frac{TP}{TP + FP}$$

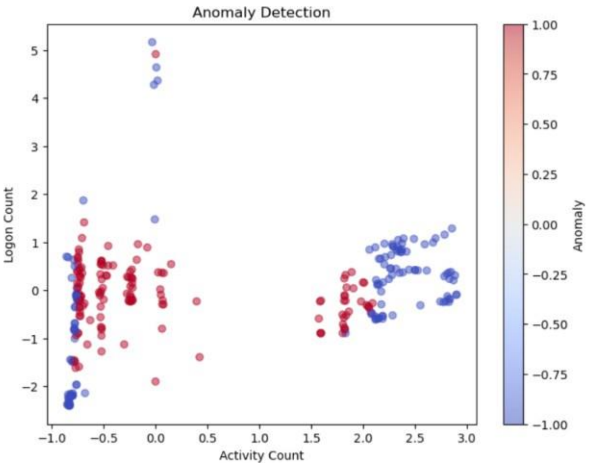
- **Recall (Sensitivity):** Percentage of real insider threats that the model accurately detected.

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

- **F1-score:** A suitable evaluation metric is provided by the harmonic mean of recall and precision.



Comparative Analysis: To ascertain the advantages of integrating several detection techniques, the outcomes of the individual models were contrasted with those of the integrated framework. This analysis involved visualizing the performance metrics and evaluating the trade-offs between different models.



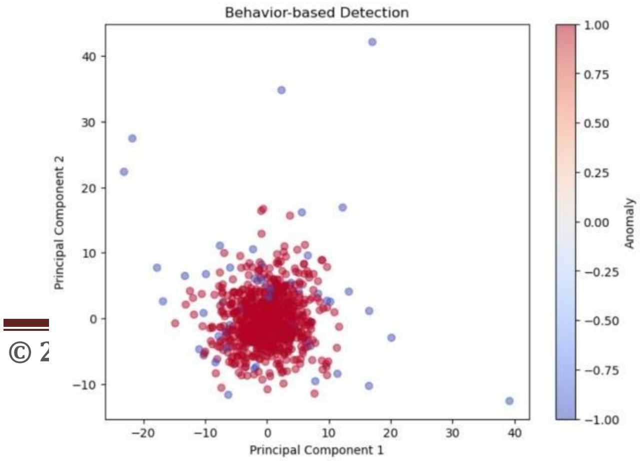


Fig. 6. Model Performance Comparison based on F1-score

E. *Results*

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

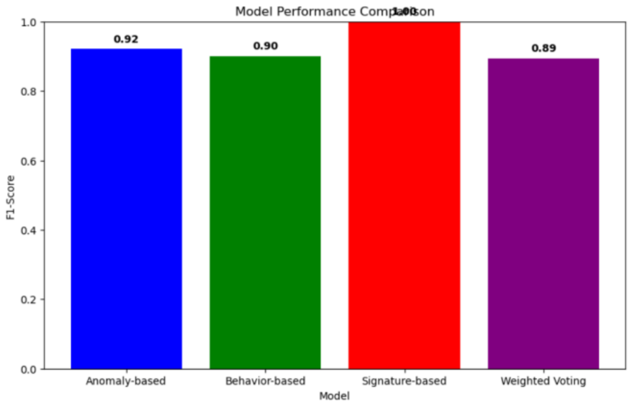
Result Compilation: A structured format containing the performance metrics for each model and the combined system was created from the experiment results. The best methods for detecting insider threats might be easily compared and identified thanks to this compilation.

We see the interpretation of each model using the relevant datasets in Fig. 7, 8 and 9. We see the results of SVM models in predicting the outliers.

Fig. 7. Anomaly Detection of the device and logon datasets.

Fig. 8. Behavior-based analysis of the user and psychometric datasets.

Fig. 9. Signature based detection of the decoy_file dataset.



V. Results Interpretation and comparisons

By combining anomaly-based, behaviour-based, and signature- based detection techniques, the integrated detection system created in this study sought to detect insider threats. The experiment's outcomes offer important new information about how successful this strategy is. We analyse these findings and contrast the effectiveness of various detection techniques.

Results Interpretation

- **Anomaly-Based Detection:**

  The anomaly-based detection module, using One-Class SVM, achieved a high precision in identifying unusual user activities that deviated significantly from established patterns. This method was particularly effective in flagging rare and unexpected behaviors, making it well-suited for uncovering novel types of insider threats. However, its reliance on detecting deviations means it might miss sophisticated threats that mimic normal behavior patterns.

- **Behavior-Based Detection:**

  The behavior-based module, leveraging psychometric data and user profiles, demonstrated robust recall, indicating its strength in identifying a broad range of potential threats based on user characteristics and behaviors. By integrating psychometric assessments, this module could correlate behavioral traits with malicious activities, enhancing its ability to predict insider threats. The dependency on accurate and comprehensive behavioral data could limit its effectiveness if such data is incomplete or outdated.

- **Signature-Based Detection:**

  Using preset patterns of known malicious activity, the signature-based detection module demonstrated great accuracy in identifying particular, well-known threat vectors. This approach produced immediate alerts for known threat signatures and was quite successful in recognizing well-known attack patterns. Its inability to identify new threats that do not fit preset signatures is its main drawback, underscoring the necessity of the signature database being updated on a regular basis.

Comparisons

The comparative analysis of the different detection methods provides a comprehensive understanding of their individual and combined performances.

Combined Approach with Weighted Voting:

The weighted voting system assigns different weights to the outputs of the three detection modules based on their relative importance and reliability. The combined score for each instance is calculated as follows:

Combined Score = ($w$anomaly x $w$anomaly)+ ($w$signature x $w$signature)+ ($w$behavior x $w$behavior)

where $w$anomaly, $w$signature and $w$behavior are the weights assigned to the anomaly-based, signature-based, and behavior-based detection modules, respectively.

In the implementation, the weights were set as follows:

- *Anomaly-based detection*: **0.5**
- *Signature-based detection*: **0.3**
- *Behavior-based detection*: **0.2**

These weights were chosen based on the performance and reliability of each module, ensuring that the most accurate and robust module has the highest influence on the final decision.

Result Classification

The combined score is then used to classify each instance as either malicious or normal. Fig. 12 shows the overview of threats. A threshold value is set to distinguish between malicious and normal activities. If the combined score exceeds the threshold, the instance is classified as malicious; otherwise, it is classified as normal. Additionally, the system can differentiate between malicious and accidental threats based on the nature and context of the detected activities.

Saving Results to CSV File

After classification, the results are saved into a CSV file for further analysis and record-keeping. The CSV file contains detailed information about each instance, including the scores from each detection module, the combined score, the final classification, and any additional metadata such as timestamps and user identifiers. This allows for easy tracking and review of detected threats, enabling prompt and effective responses to both malicious and accidental threats. Table. 1 shows the detected threats by the system.

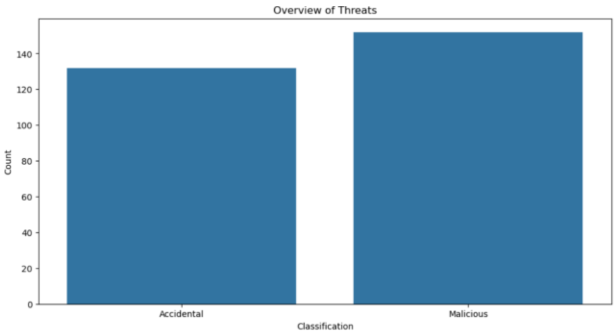
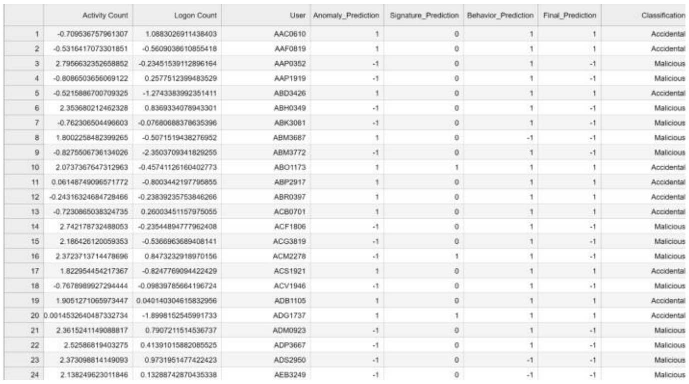


Fig. 10. Overview of Threats detected as accidental and malicious threats

TABLE I DETECTED



	Activity Count	Logon Count	User	Anomaly_Prediction	Signature_Prediction	Behavior_Prediction	Final_Prediction	Classification
1	-0.709536757961307	1.0883026911438403	AAC0610	1	0	1	1	Accidental
2	-0.5316417073301851	-0.5609038610855418	AAF0819	1	0	1	1	Accidental
3	2.7956632352658852	-0.23451539112896164	AAP0352	-1	0	1	-1	Malicious
4	-0.8086503656069122	0.2577512399483529	AAP1919	-1	0	1	-1	Malicious
5	-0.5215886700709325	-1.2743383992351411	ABD3426	1	0	1	1	Accidental
6	2.353680212462328	0.8369334078943301	ABH0349	-1	0	1	-1	Malicious
7	-0.762306504496603	-0.07680688378635396	ABK3081	-1	0	1	-1	Malicious
8	1.8002258482399265	-0.5071519438276952	ABM3687	1	0	-1	-1	Malicious
9	-0.8275506736134026	-2.3503709341829255	ABM3772	-1	0	1	-1	Malicious
10	2.0737367647312963	-0.45741126160402773	ABO1173	1	1	1	1	Accidental
11	0.06148749096571772	-0.8003442197795855	ABP2917	1	0	1	1	Accidental
12	-0.24316324684728466	-0.23839235753846266	ABR0397	1	0	1	1	Accidental
13	-0.7230865038324735	0.26003451157975055	ACB0701	1	0	1	1	Accidental
14	2.742178732488053	-0.23544894777962408	ACF1806	-1	0	1	-1	Malicious
15	2.186426120059353	-0.5366963689408141	ACG3819	-1	0	1	-1	Malicious
16	2.3723713714478696	0.8473232918970156	ACM2278	-1	1	1	-1	Malicious
17	1.822954454217367	-0.8247769094422429	ACS1921	1	0	1	1	Accidental
18	-0.7678989927294444	-0.09839785664196724	ACV1946	-1	0	1	-1	Malicious
19	1.9051271065973447	0.040140304615832956	ADB1105	1	0	1	1	Accidental
20	0.0014532640487332734	-1.8998152545991733	ADG1737	1	1	1	1	Accidental
21	2.3615241149088817	0.7907211514536737	ADM0923	-1	0	1	-1	Malicious
22	2.52586819403275	0.41391015882085525	ADP3667	-1	0	1	-1	Malicious
23	2.373098814149093	0.9731951477422423	ADS2950	-1	0	-1	-1	Malicious
24	2.138249623011846	0.13288742870435338	AEB3249	-1	0	-1	-1	Malicious

CONCLUSION AND FUTURE SCOPE

The development of the Integrated Insider Threat Detection System (IITDS) represents a significant advancement in enhancing the security of organizational IT infrastructures. This strategy seeks to greatly increase the precision and effectiveness of insider threat identification by combining behavioral analysis, anomaly detection and signature-based detection techniques. The creation of an extensive system that painstakingly prepares and processes data, trains models in a methodical manner, and carefully assesses their performance are among the major accomplishments. This methodical process improves feature engineering, eliminates anomalies, and guarantees data integrity—all essential for the system's ability to identify insider threats. Insights gleaned from data preparation underscore the critical importance of high-quality data for effective threat detection. By blending rule-based decisions with machine learning predictions, the method enhances decision-making processes. Specific rules tailored to user behaviors and strict adherence to security standards bolster the method's efficacy in identifying and preventing insider threats. Looking forward, there are promising avenues to enhance the system's efficacy and versatility. Advancements in machine learning algorithms will enable finer granularity in threat detection, reducing false positives and improving detection rates. Exploring advanced techniques such as ensemble methods holds potential for further enhancing predictive analytics capabilities. Developing robust mechanisms for real-time activity monitoring and rapid response to emerging threats will be pivotal. Adaptive learning frameworks that facilitate dynamic model updates and responsive strategies are essential for maintaining proactive threat prevention measures. The integration of explainable AI methodologies will enhance the transparency and interpretability of model decisions. Providing actionable insights to stakeholders promotes informed decision-making, fosters regulatory compliance, and builds stakeholder trust. Beyond the cybersecurity domain, the system's adaptability extends to applications in healthcare, e-commerce, and financial sectors. Its proven efficacy in threat detection positions it as a valuable tool for enhancing operational resilience, customer trust, and regulatory adherence across diverse industries.

ACKNOWLEDGMENT

Required resources are provided by the Department of CSE(CS), Institute of Aeronautical Engineering, Hyderabad, India for this paper's research study and related work.

References

[1] S. Natarajan, V. P. Vemuri, S. H. Krishna, Y. M. Reddy, P. Gundawar and S. Lakhanpal, "Prediction Analysis of AI Adoption in Various Domain Using Random Forest Algorithm," *2024 International Conference on Communication, Computer Sciences and Engineering (IC3SE)*, Gautam Buddha Nagar, India, 2024, pp. 1537-1541, doi: 10.1109/IC3SE62002.2024.10593362.

[2] R. C. Dodge Jr., A. J. Ferguson and D. M. Cappelli, "Introduction to Insider Threat Modeling, Detection, and Mitigation Minitrack," 2013 46th Hawaii International Conference on System Sciences, Wailea, HI, USA, 2013.

[3] Y. Guerbai, Y. Chibani and B. Hadjadji, "The effective use of the One-Class SVM classifier for reduced training samples and its application to handwritten signature verification," 2014 International Conference on Multimedia Computing and Systems (ICMCS),
Marrakech, Morocco, 2014, pp. 362-366

[4] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” in Proc. Int. Conf. Learn. Represent., 2015, pp. 1–15.

[5] M. Abadi et al. (2015). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. [Online]. Available: https://www.tensorflow.org/

[6] M. L. Collins et al., “Common sense guide to mitigating insider threats, fifth edition,” The CERT Insider Threat Center, Tech. Rep., 2016, CMU/SEI-2015-TR-010.

[7] L. Lin, S. Zhong, C. Jia and K. Chen, "Insider Threat Detection Based on Deep Belief Network Feature Representation," 2017 International Conference on Green Informatics (ICGI), Fuzhou, China, 2017.

[8] M. Kim, K. Kim and H. Lee, "Development trend of insider anomaly detection system," 2018 20th International Conference on Advanced Communication Technology (ICACT), Chuncheon, Korea (South), 2018.

[9] C. H. Park, "Anomaly Pattern Detection on Data Streams," 2018 IEEE International Conference on Big Data and Smart Computing (BigComp), Shanghai, China, 2018, pp. 689-692

[10] Crowd Research Partners, “2018 insider threat report,” CA Technologies, Tech. Rep., 2018, https://crowdresearchpartners.com/ insider-threat-report.

[11] W. Meng, K.-K. R. Choo, S. Furnell, A. V. Vasilakos, and C. W. Probst, “Towards bayesian-based trust management for insider attacks in healthcare software-defined networks,” IEEE Trans. Netw. Service Manag., vol. 15, no. 2, pp. 761–773, Jun. 2018.

[12] Y. Zhao, Z. Nasrullah, and Z. Li, “PyoD: A Python toolbox for scalable outlier detection,” J. Mach. Learn. Res., vol. 20, no. 96, pp. 1–7, 2019.

[13] I. Homoliak, F. Toffalini, J. Guarnizo, Y. Elovici, and M. Ochoa, “Insight into insiders and IT: A survey of insider threat taxonomies, analysis, modeling, and countermeasures,” ACM Computing Surveys, vol. 52, no. 2, pp. 30:1–30:40, Apr. 2019.

[14] L. Liu, C. Chen, J. Zhang, O. De Vel, and Y. Xiang, “Insider threat identification using the simultaneous neural learning of multi-source logs,” IEEE Access, vol. 7, pp. 183162–183176, 2019.

[15] E. H. Budiarto, A. Erna Permanasari and S. Fauziati, "Unsupervised Anomaly Detection Using K-Means, Local Outlier Factor and One Class SVM," 2019 5th International Conference on Science and Technology (ICST), Yogyakarta, Indonesia, 2019, pp. 1-5

[16] D. C. Le and N. Zincir-Heywood, “A frontier: Dependable, reliable and secure machine learning for network/system management,” J. Netw. Syst. Manag., vol. 28, pp. 827–849, Jan. 2020.

[17] Q. Ma and N. Rastogi, "DANTE: Predicting Insider Threat using LSTM on system logs," 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Guangzhou, China, 2020.

[18] Dong Chen and Yang Wu. 2024. Research on the use of communication big data and AI artificial intelligence technology to construct telecom fraud prevention behavior portrait. Int. Dec. Tech. 18, 3 (2024), 2589–2605. https://doi.org/10.3233/IDT-240386

[19] Mahboubi, A., Luong, K., Aboutorab, H., Bui, H. T., Jarrad, G., Bahutair, M., Camtepe, S., Pogrebna, G., Ahmed, E., Barry, B., & Gately, H. (2024). Evolving techniques in cyber threat hunting: A systematic review. *Journal of Network and Computer Applications*, *232*, 1-34. Article 104004. Advance online publication. https://doi.org/10.1016/j.jnca.2024.104004

[20] D. C. Le and N. Zincir-Heywood, "Anomaly Detection for Insider Threats Using Unsupervised Ensembles," in *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1152- 1164, June
2021.

[21] X. Sun, Y. Wang and Z. Shi, "Insider Threat Detection Using An Unsupervised Learning Method: COPOD," 2021 International Conference on Communications, Information System and Computer Engineering (CISCE), Beijing, China, 2021

[22] G. Padmavathi, D. Shanmugapriya and S. Asha, "A Framework to Detect the Malicious Insider Threat in Cloud Environment using Supervised Learning Methods," *2022.*

[23] Y. Li and Y. Su, "The Insider Threat Detection Method of University Website Clusters Based on Machine Learning," *2023 6th International Conference on Artificial Intelligence and Big Data (ICAIBD)*, Chengdu, China, 2023.

[24] 2023. Proceedings of the 2023 8th International Conference on Information and Education Innovations. Association for Computing Machinery, New York, NY, USA.

[25] Mohammad Amiri-Zarandi, Hadis Karimipour, Rozita A. Dara, A federated and explainable approach for insider threat detection in IoT, Internet of Things, Volume 24, 2023, 100965, ISSN 2542-6605, https://doi.org/10.1016/j.iot.2023.100965.

[26] F. R. Alzaabi and A. Mehmood, "A Review of Recent Advances, Challenges, and Opportunities in Malicious Insider Threat Detection Using Machine Learning Methods," in *IEEE Access*, vol. 12, pp. 30907- 30927, 2024

[27] F. R. Alzaabi and A. Mehmood, "A Review of Recent Advances, Challenges, and Opportunities in Malicious Insider Threat Detection Using Machine Learning Methods," in *IEEE Access*, vol. 12, pp. 30907-30927, 2024, doi: 10.1109/ACCESS.2024.3369906.