Enhanced Insider Threat Detection Using Machine Learning and Hayabusa-based User Behaviour Analytics

Abhinav Pandey
Dept. of Computer Science and Engineering (Cyber Security)
Dayananda Sagar College of Engineering Bangalore, India
abhinav04.ap@gmail.com

Arantha Shreya Gogoi
Dept. of Computer Science and Engineering (Cyber Security)
Dayananda Sagar College of Engineering Bangalore, India
aranthashreya@gmail.com

Divyanshi Chaudhary
Dept. of Computer Science and Engineering (Cyber Security)
Dayananda Sagar College of Engineering Bangalore, India
chaudharydivyanshi238@gmail.com

Tanu Rajput
Dept. of Computer Science and Engineering (Cyber Security)
Dayananda Sagar College of Engineering Bangalore, India
ttanurajput2568@gmail.com

Mangala L
Dept. of Computer Science and Engineering (Cyber Security)
Dayananda Sagar College of Engineering Bangalore, India
mangalashresta@gmail.com

***Abstract*—This research paper presents an intelligent hybrid framework that uses forensic log analysis along with machine learning-based user behavior modeling for the detection of insider threats. This paper explores how parsing of Windows event logs and forensic timeline construction, along with using machine learning models for behavioral anomaly detection based on mouse movements, typing speed, click rhythms, keystroke latency, mouse-movement trajectories, and session-time patterns, can be used for the detection of insider threats. Isolation Forest, Random Forest, SVM, and LSTM are some models used to detect deviations from normal user behavior. This framework not only produces an objective anomaly score but also provides forensic evidence to establish accountability. This is done by correlating event timelines from Windows logs with behavioral patterns derived from the ML model. The paper shows how integration of AI with cyber forensics can help in the detection of insider threats.**

***Index Terms*—Insider Threat Detection, User Behavior An- alytics (UBA),Digital Forensics, Machine Learning, Anomaly Detection, Behavioral Biometrics, Credential Misuse, Log Anal- ysis, Isolation Forest, Random Forest, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), Event Correlation, Cybersecurity, Windows Event Logs.**

I. Introduction and Motivation

Insider threats are among the most dangerous types of attacks, as they come from trusted users inside the organization who already possess valid authentication credentials. Because trusted users have legal access, identifying malicious insider activity is more complex than outsider threats. Trusted users, by actions that seem legitimate, can easily act in an illegitimate manner within normal access protocols. Auxiliary perimeter-based security tools - such as firewalls and signature-based IDS only consider external threat behaviour, and therefore are of little use for insider misuse [1], [2].

Recent research strongly highlights that combining Ma- chine Learning (ML) techniques with User Behaviour Anal- ysis (UBA) provides a more reliable, data-driven strategy for insider threat detection [3], [4]. ML algorithms will constantly learn patterns based on historical logs, including log-in sequences, file access behavior, command patterns, process behavior, and network flows. If the learned statistical patterns deviate from that user's normal profile, it suggests that there may be an insider anomaly. Behavioral biometrics, like keystroke rhythm, timing patterns, and interaction frequency, assist in differentiating between legitimate users and users that are masquerading [5], [6].

The growing trend of ML-based behaviour analytics is due to modern enterprise trends - remote working, cloud access, BYOD environments and acceleration of privileges - all contribute to making insider attacks increasingly possible and

much more damaging [7], [8]. Large organizations generate millions of events each day making it impractical to mon- itor human behaviour manually. Consequently, ML models (e.g. Random Forest, SVM, LSTM, GRU, Isolation Forest, XGBoost, hybrid ensembles) are now widely being applied to automatically classify suspicious behaviour and minimize false positives [9], [10], [11]. Additionally, deep learning (e.g. LSTM, GRU) techniques have been shown to capture sequential changes in behaviour over time which aids in detecting slow moving insider threats [12]. Hybrid or ensemble architectures have also shown improved detection accuracy by using multiple ML classifiers instead of relying on a single model [13], [14].

As a result, the objective of this survey is to investigate how machine learning models augmented with behaviour-based analytics can advance the efficiency of insider threat detection, reduce false positives, and increase the trustworthiness of early-warning systems. By evaluating and reviewing the latest research, this paper will compare a number of machine learn- ing (ML) methods, investigate differences in performance, highlight their limitations, and recommend future real-time insider threat detection framework approaches.

II. LITERATURE REVIEW/ RELATED WORK

Many approaches have been put out over the years to improve detection efficiency and accuracy in the field of net- work intrusion detection, which has been the subject of much research. To detect malicious activity, traditional Network In- trusion Detection Systems (NIDS) mostly use anomaly-based and signature-based techniques. Signature based systems use a predefined set of attack signatures to match patterns in network traffic. . Identifying authorized users who are harming the organization while they are trusted is the most difficult cyber security challenge . These systems are limited in their ability to detect new or zero-day attacks that do not yet have associated signatures, despite being successful in detecting established threats. Anomaly-based systems, on the other hand, keep an eye out for variations from typical traffic patterns and may be able to identify attacks that have not yet been noticed. They frequently generate a large number of false positives, though, which can overwhelm security analysts and lower the system's overall effectiveness [8]. Insider threats are a real problem that is emerged form legitimate users who already possess autho- rized access to the sensitive information or systems, it makes difficult to distinguish these internal attacks from the malicious activities[1]. Some of the conventional security systems con- sists of the mechanisms like firewalls, antivirus programs and signature based intrusion detection system (IDS), they detect known patterns, but for the subtle change in the behaviour they fail[2]. Earlier insider threat detection systems relied mostly on rule based or signature based approaches that depend upon static threshold or manually defined behavioral pro- files.Initially, insider threat detection systems that were early in the game depended on incorporated rule-based or signature- based techniques that required fixed thresholds or manually defined behavior profiles. These systems, while capable of detecting known attack patterns, were not flexible and couldn't handle new or altered threat behaviors[3]. To overcome these challenges, researchers started to focus on machine learning (ML) approaches. Such methods are capable of continuously learning behavior patterns and detecting variations instantly. Through the inspection of past users' activities data, the ML models can surface intricate patterns and relationships that indicate potential insider actions without the need for explicit rules. Insider threats are a real problem that is emerged form legitimate users who already possess authorized access to the sensitive information or systems, it makes difficult to distin- guish these internal attacks from the malicious activities[1]. Some of the conventional security systems consists of the mechanisms like firewalls, antivirus programs and signature based intrusion detection system (IDS), they detect known patterns, but for the subtle change in the behaviour they fail[2]. Early warning tools for insider threats mostly used fixed rules or set-in-stone signs, often based on rigid limits or hand- crafted behavior checklists. Although such methods worked well spotting known attack types, they couldn't adjust easily and had trouble catching new or shifting risky actions [3].To fix gaps studies focused on training ML models, which instead of relying on present conditions, trains from the previous user logs. It reveals what can be potential threat from internal risks. Methods like decision tree, SVMs, Random Forest, or Logistic Regression are the supervised learning models they are applied to get regular behaviour of user that can be potential threat [1],[8]. They work well in deciding the known attack patterns and catching them, it can be used in crime analysis also. Still, these approaches require a lot of data and and it becomes tough because of privacy issues and sensitive information leaks[5]. To tackle the issues with the tagging data, methods like anomaly spotting and learning without labels is used. In the paper Mehmood and team[7] used grouping along the stats-based outliner detector to spot actions in how users behave. These systems figure out normal activities if anything seems off-pattern then it is suspicious. Deep learning can pick up tricky patterns instead of systems like RNNs and LSTMs. There are further studies conducted like adding group- based tools or gradient runs were effective, some conducted analysis for windows event logs, showing how organizational tracking. Based on the performance comparison, it can be observed that deep learning sequence models (e.g., LSTM, GRU) out perform traditional ML approaches because they are able to capture user activity patterns across time. For example, ensemble models(e.g., XG Boost) generally follow with higher accuracy in predicting insider threats, as they are able to combine multiple decision factors into a collective model. ML approaches such as SVMandRandomForestmodels have lower accuracy, but still provide reasonable predictions on structured log data, though struggle to react to ongoing variational behaviour. Unsupervised based models can assist researchers when labelled datasets to test on are scarce, but often have low accuracy.

Based on the performance comparison, it can be observed that deep learning sequence models (e.g., LSTM, GRU) out-

TABLE I
SUMMARY OF KEY RESEARCH PAPERS IN INSIDER THREAT DETECTION

No.	Author(s) & Year	Title / Core Contribution	Limitation / Research Gap
1	Amit Hariyani, Jaimin Undavia, Nilay Vaidya, Atul Patel (2022)	*Forensic Evidence Collection From Windows Host Using Python Based Tool* — A Python tool to collect forensic artifacts like Windows Event Logs, Registry values, and System logs.	Platform-specific for Windows. Real-time responses were not available.
2	Manju A., Mazidah Puteh, Subha R. (2025)	*Insider Threat Detection using Machine Learning Models for User Behavior Anal- ysis* — Insider threat detection using clus- tering, anomaly detection, and deep learning for BYOD and cloud contexts.	Real-time detection and multi-platform generaliza- tion were not addressed.
3	Hanay Almomani et al. (2024)	*Proactive Insider Threat Detection Using Facial and Behavioral Biometrics* — Combined eye movement, facial expression, and stress indicator analysis for enhanced detection.	Large-scale deployment was not possible due to ethical/privacy concerns, implementation complications, and limited dataset diversity.
4	Favour Femi-Oyewole, Victor Osamor, Daniel Okunbor (2024)	*Survey on Predictive Algorithms to Detect Insider Threat on a Network Using Different Combination of ML Algorithms* — PRISMA-based survey of ML/DL algorithms (SVM, RNN) for insider threat prediction.	Scope limited to a literature review; no new predictive model proposed; only studies up to 2024 included.
5	Rahat Mehmood, Priyanka Singh, Zoe Jeffery (2024)	*Unsupervised Learning for I Prediction: A Behavioral Anal* — Introduced an unsupervise using the CERT dataset.	
6	Pannada Siva Satya Prasad, Sasmita K. Nayak, M. Vamsi Krishna (2024)	*Enhancing Insider Threat D Machine Learning Technique* dom Forest and AdaBoost o achieving 97% accuracy.	
7	Manikandan S.P. et al. (2024)	*Detecting Insider Threats in Using Machine Learning* — using Decision Trees and sion optimized by Grid Search	
8	D. Sridevi, L. Kannagi, Vivekanandan G., S. Revathi (2023)	*Detecting Insider Threats U Learning and Deep Learning* Hybrid ML-DL architecture Net) achieving 96.3% accurac	
9	Anant Wairagade, Sumit Ranjan (2025)	*User Behavior Analysis for Detection: A Comparative S Learning Algorithms* — Comp UBA methods for threat detec	
10	Yi Zhang (2022)	*Behavioral Biometrics Identif on Machine Learning Meth* Decision Trees, SVM, and celerometer data for user ID.	

perform traditional ML approaches because they are able to capture user activity patterns across time. For example, ensemble models (e.g., XGBoost) generally follow with higher accuracy in predicting insider threats, as they are able to combine multiple decision factors into a collective model. ML approaches such as SVM and Random Forest models have lower accuracy, but still provide reasonable predictions on structured log data, though struggle to react to ongoing variational behaviour. Unsupervised based models can assist researchers when labelled datasets to test on are scarce, but often have low accuracy[16][17][18].

III. LIMITATIONS OF EXISTING APPROACHES

- **Behavioural Drift:** Evolving user behaviour is not addressed effectively. Results from models trained on his- torical datasets tend to degrade over time as normal user behaviour changes and evolves. This lack of adaptability causes older models to miss new threat patterns and user habits [1], [6].

Dependence on Data Quality and Completeness: Qual- ity of the dataset impacts the results of the framework greatly; missing, corrupted, or noisy records can cause misclassification or missed detections [8], [12].

- **Limited Real-Time Adaptability:** Creation of timelines and correlation can cause latency, which is an obstacle in generating real-time responses [8], [9].
- **High Computational Overhead and Scalability:** Running multiple models requires a large amount of process- ing power, which may require distributed processing to implement at a large scale. [2], [10].
- **Interpretability:** The results from deep learning models are opaque. It is extremely hard to tell why and how a certain alert was raised. [7], [11].

TABLE II
PERFORMANCE COMPARISON OF AI MODELS FOR INSIDER THREAT DETECTION

ML Model / Approach	Type	Detection Accuracy	Strengths	Weaknesses
Random Forest (RF)	Supervised ML	88–92%	Interpretable, good for tabular logs	May overfit large noisy logs
Support Vector Machine (SVM)	Supervised ML	84–89%	Works well on small datasets	Not scalable for high-dimensional UBA features
Isolation Forest	Unsupervised ML	78–85%	No need fo mining	
LSTM (Deep Learning)	Sequential DL	89–93%	Learns ten	
GRU (Deep Learning)	Sequential DL	88–92%	Faster than	
XGBoost Ensemble	Hybrid Ensemble	90–94%	Best gener	
AutoEncoder	Unsupervised DL	85–90%	Reconstruc	

- **Limited Generalizability of Training Data:** Reliance on synthetic behavior datasets allows for limited general- ization, as synthetic datasets do not capture all real-world scenarios. [5], [13].
- **Privacy and Ethical Concerns:** Capturing behavioral data may raise legal and ethical issues. This requires strict data handling, minimization, and anonymization measures [4], [13].
- **Limited Multi-Platform Coverage:** Capturing behav- ioral data may raise legal and ethical issues. This requires strict data handling, minimization, and anonymization. [4], [13]. • Limited multi-platform coverage: Most of the forensics components are Windows-focused and will re- quire work to extend functionality to other platforms. [8].

IV. IDENTIFIED RESEARCH CHALLENGES

Even though great progress has been made in behavioural analytics and anomaly-based detection, there are still some difficulties in mitigating insider threats.

A. Data Quality and Imbalance

Lack of quality data and reliance on synthetic datasets like CERT cause problems in model training as they do not repre- sent real-world scenarios completely[1],[2],[5]. Since data on insider threat is negligible when compared to Legitimate user biases towards typical behaviour is created in model training [6]. It is found that if the attack frequency is less than 5 percent models incur recall loss[6][7].

B. Behavioral Drift and Context Sensitivity

User habits and behaviour change due to various factors, causing changes in behavioural biometrics [2][3]. Incorporat- ing these changes in the model training is a huge challenge as most data sets are static in nature[4].

C. Feature Engineering and Correlation

Parsing of useful information from log files is a key step in insider detection, but Extraneous features or noisy data may cause classification tools to fail. Data like keyboard typing rhythm and movement of the cursor of each user are very hard for models to detect correctly[3][5]. Some papers suggest that using PCA along with autoencoders may help in merging features[6][10].

D. Making Sense of Results So Analysts Feel Confident

The results from deep learning models are opaque. It is extremely hard to tell why and how a certain alert was raised. [7], [11]. Deep learning models like LSTM are highly accurate but do not provide transparency in how the result was achieved[7][10].

E. Evidence from crime scenes gets mixed into investigations

Many of the machine learning frameworks do not incor- porate cyber-investigation methods. Due to the absence of links between system logs and behaviour analysis data, the trustworthiness of evidence reduces greatly. This also slows down investigation probes.[8] reduces analyst verification time from hours to minutes.

V. CONCEPTUAL METHODOLOGY

The fig.1 shows the steps taken to apply selected machine learning algorithms, for the first the data collected is done, data pre-processing, file aggregation, feature extraction, feature selection, Algorithm selection and model development, Data aggregation And model training and evaluation are in the steps. In the fig.2 it's the implementation of Hayabusa tool, how it helps in log parsing and timeline generation, it cleans the data based on sigma rules. In the feature engineering along with machine learning modeling, traits like how often someone logs in, what commands they use, or odd access times get pulled out then studied using mixed ML methods. Later on , these investigation tools helps system boost the threat detection model and easier tracking across the company networks.

VI. VISUALIZATION AND FORENSIC REPORTING

To improve the interpretation and understanding of results, using visualisation can help analysts greatly. An interface that can visualise algorithmic outputs can help analysts to reconstruct events, detect anomalies and generate evidence.

A. Dashboard Design

Traditional setups use dashboards and risk maps [1], [5], [10]. This can be proved by adding:

- **Heatmaps:** To visualise the hourly anomaly intensities of users

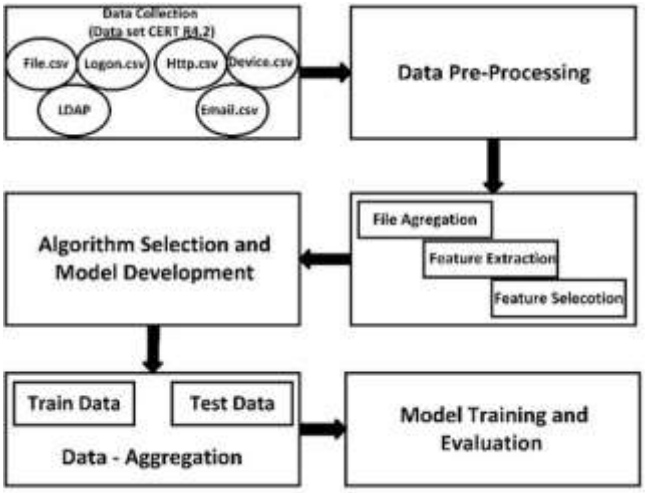

Fig. 1. Steps taken to apply selected machine learning algorithms.

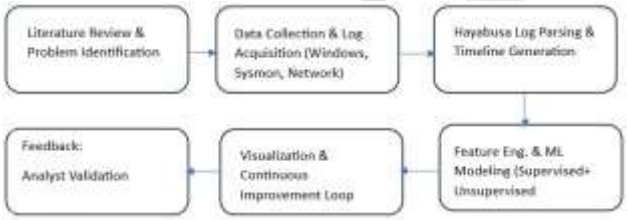

Fig. 2. End-to-End Workflow for the Proposed Framework.

- **Radar charts:** These compare the normal user behaviour data, like typing and mouse movements, to anomalous behaviour
- **Process trees:** Rebuild parents-child relations using Hayabusa log data

B. Timeline Correlation via Hayabusa

Hayabusa parses important data such as Event ID, User SID, process name, and Timestamp [8]. These data can be synchronised with the anomaly flagged by the ML model, resulting in the creation of clear timelines and distinguishing real activity from impersonations [5], [7], [9].

C. Automated Evidence Reports

The reporting section puts out:

1) Anomalous user and session IDs are marked.
2) Events and processes are correlated.
3) Model confidence scores are combined with risk levels.

Studies show that this kind of visualization greatly reduces the time spent by analysts on verification of results [7], [8].

VII. Discussions and Future Directions

The current scenario regarding insider threat requires fore- cast techniques along with audit trails. Current ML methods produce good results in flagging odd users, but are unable to provide proof or reasons as to why the user was flagged after the breaches occur. [1]– [15].

A. Integrative Discussion

Traditional methods to detect insider threats completely depend on ML models to flag anomalous behaviour. Even though ML methods produce good results in flagging odd users, but are unable to provide proof or reasons as to why the user was flagged after the breaches occur [1], [2], [6]. The problem with using ML models is that they are unable to tie the flagged behaviour with actual system- level events, making follow-up tough. To fix this issue, our projects create a correlation between the parsed Windows logs and the flagged behaviours obtained through the ML model. Through Hayabusa, we are able to obtain things like Event ID, Process ID, and timestamps [8]; which are then correlated with anomalous behaviours like strange typing rhythms, cursor paths, or strange app launch habits obtained through the ML model. Whenever a behaviour is flagged by the ML model, we can use data obtained from Hayabusa to check if the flagged behaviours line up with sketchy actions like strange sign-ins, sudden admin access jumps, or weird program runs. This not only increases the model's accuracy but also helps in building proof trails that match warnings to actual clues. This framework fixes what pure ML-based frameworks can't handle [3], [5], [7], [9]by providing background context and logging who did what during the breach. Some test shows that this combined setup is able to reduce the false positives greatly [7], [9]. Therefore, by combining different methods instead of relying on one method, we are able to turn raw results into insights that are backed by facts, which can act as evidence or after-the-fact analysis.

B. Operational Implications

1) **Accelerated Forensic Investigation:** By combining different methods, we are able to decrease the time spent on analysis. Rather than going through the entire logs, we can just see event chains that link up with odd behaviours. This greatly reduces the burden of the investigating team.
2) **Evidence-Based Attribution:** By matching odd be-haviour with event logs, we are able to create evidence that is based, the results explanations are not vague but have solid backing by knowing who did what during the breach.
3) **Seamless SOC Integration:** Since Hayabusa creates standard JSON and CSV files, our projects can easily be integrated into enterprise-level tracking setups like Splunk or QRadar.
4) **Compliance and Audit Readiness:** Organisations can maintain audit logs and preserve data integrity and user data at the same time, as the setup's reporting layer aligns with ISO 27037 and GDPR principles for evidence handling.

C. Theoretical and Research Implications

Since our project is able to produce a correlation between the anomalous behaviour and flagged user, we are able to grow explainable forensics intelligence. This not only spots

odd patterns like traditional systems but is able to produce an explanation for why the behaviour was flagged. By tracing back to the events that happened, the flagged behaviour can be explained to a great extent. This correlation creates great learning loops, which allow for greater accuracy by machines. Therefore, by combining clues and actions of the user, we are able to create a smarter digital forensics tool.

D. Future Research Directions

1) **Adaptive Learning for Behavioral Drift:** Since the behaviour of legitimate users also varies and changes due to various factors, a system that can take this variable into account will lead to a reduction in false positives. This may be done using Adaptive Isolation Forest or tools that detect pattern shifts as they happen [2], [4], [6].

2) **Cross-Platform Forensic Generalization:** Since the logs that are parsed are Windows-based based we can further extend the functionality by adding support to other systems like Linux and macOS, which can increase the generalisation of the framework.[[8].

3) **Explainable AI (XAI) for Model Transparency:** Tools like SHAP or LIME can be integrated to show how the decisions were made, making the analysts more confident in the results [11].

4) **Federated and Privacy-Preserving Learning:** Differ- ent methods can be implemented to secure the privacy of users, such as decentralised processing and training models without centralising sensitive data. [12].

5) **Graph Neural Networks for Relationship Mapping:** We can create relational dependencies between users, processes, and devices by using Graph Neural Networks (GNNs), which help in the detection of collusive or multi-user insider scenarios [10], [13].

6) **Synthetic–Real Hybrid Datasets:** Creation of hybrid datasets that blend synthetic logs with real-world data can allow for training more accurate models [1], [6].

7) **Behavioral Provenance and Intent Modeling:** We can incorporate pattern analysis to guess what someone might do, which can help in spotting benign careless users and differentiating intentional threats [3], [14].

E. Long-Term Vision

Creation of fully autonomous forensic investigation tools that not only detect suspicious behaviour but also create complete incident timelines with minimum human interaction. Self-running FASOCs can be created by integrating tools, machine learning, event timelines and clear visuals. Future sys- tems can adapt to track actions across systems to find breaches as well as describe them as they happen. Additionally, this system creates greater accountability by not only detecting issues but tracing them back so that each warning from the system rests on solid analysis plus can stand up under close review. Going forward, responsibility, clear processes, along linking different fields together should be the main focus of studies for building reliable security systems.

VIII. CONCLUSION

The report explained the advancements in detecting in- sider threats using machine learning algorithms to analyse behaviours. Some of the problems identified in existing en- terprise environments included high false positive rates, dif- ficulty in distinguishing abnormal versus normal behaviour, and rule-based security controls being unable to detect cre- dential misuses by legitimate users within the designated ac- counts. Machine learning-based monitoring systems, founded on behavioural profiling, can provided targeted and scalable solutions to mitigate the issues. Monitoring systems can con- tinuously profile user behaviour patterns, and will be able to identify deviations much earlier than when the abnormal be- haviour progresses to insider activity—specifically, imperson- ation behaviour occurring. The outcome of machine learning- based anomaly detection combined with behavioural analytics can help bridge the fixed, static security controls paradigms to the fluid modern insider threat by providing for greater effectiveness in risk identification, greater mechanisms of automated detection, and more operationally practical security controls.

REFERENCES

[1] M. A. Manju, M. Puteh, and S. R. Subha, “Insider Threat Detection using Machine Learning Models for User Behavior Analysis,” *Proc. 5th Int. Conf. Expert Clouds and Applications (ICOECA)*, 2025.

[2] A. Wairagade and S. Ranjan, “User Behavior Analysis for Cyber Threat Detection: A Comparative Study of Machine Learning Algorithms,” *Proc. 13th Int. Symp. Digital Forensics and Security (ISDFS)*, 2025.

[3] A. Hariyani, J. Undavia, N. Vaidya, and A. Patel, “Forensic Evidence Collection from Windows Host using Python-Based Tool,” *Proc. 4th IEEE Int. Conf. Cybernetics, Cognition and Machine Learning Applica- tions (ICCCMLA)*, 2022.

[4] H. Almomani, A. K. Al-Azzam, and N. Almulhem, “Proactive Insider Threat Detection Using Facial and Behavioral Biometrics,” *Proc. 7th Int. Conf. on Advanced Computer and Information Technology (ACIT)*, IEEE, 2024.

[5] M. A. Khan, A. N. Khan, and A. I. Khan, “Behavioral Biometrics Iden- tification Based on Machine Learning Methods,” *IEEE Conf. Machine Learning and Knowledge Engineering (MLKE)*, 2022.

[6] F. Femi-Oyewole, V. Osamor, and D. Okunbor, “Survey on Predictive Algorithms to Detect Insider Threat on a Network Using Different Combination of Machine Learning Algorithms,” *Proc. IEEE Sustainable Energy-Based Solutions for Smart Digital Governance (SEB4SDG)*, 2024.

[7] R. Mehmood, A. Khan, M. A. Rahman, and H. Gani, “Unsupervised Learning for Insider Threat Prediction: A Behavioral Analysis Ap- proach,” *Proc. IEEE TrustCom*, 2023.

[8] R. Nasir, M. S. Shafiq, and F. Kausar, “Detecting Insider Threats in Cybersecurity Using Machine Learning,” *IEEE Access*, vol. 10, pp. 11284–11296, 2022.

[9] S. R. Eguavoen and E. Nwelih, “HSML-ITD: Hybrid Supervised Ma- chine Learning Framework for Insider Threat Detection,” *Quantum Journal of Engineering Science and Technology*, vol. 2, no. 1, pp. 12–22, 2025.

[10] J. Dixit, K. Alshehri, and R. Kim, “Deep Clustering and LSTM for Insider Threat Detection,” *Proc. IEEE Int. Conf. on Artificial Intelligence and Security (ICAIS)*, 2023.

[11] S. Dhaygude, T. Wei, and Z. Zhang, “Graph-Based and Ensemble Learning Methods for Detecting Insider Threats,” *Proc. IEEE Conf. Information Security and Privacy (ICISP)*, 2024.

[12] P. Bin Sarhan and H. Altwaijry, “Predictive Models Employing LSTM and GRU for Insider Threat Detection,” *IEEE Transactions on Informa- tion Forensics and Security (TIFS)*, vol. 18, pp. 744–756, 2024.

[13] W. Saini, M. Sakthivelu, and K. Zala, “Hybrid Random Forest and XGBoost Ensemble for Insider Threat Detection,” *Proc. CSE-CIC IDS Challenge Workshop*, IEEE, 2023.

[14] A. Apruzzese, M. Colajanni, and D. Ferri, "Generative Adversarial Networks for Insider Threat Simulation and Detection," *Proc. IEEE Workshop on Cyber Threat Intelligence (WCTI)*, 2024.
[15] M. Ricci *et al.*, "Hybrid Keys in Practice: Combining Classical, Quantum, and Post-Quantum Cryptography," *Proc. IEEE Symp. Security and Privacy*, 2024.
[16] C. V. Balaji, A. Bharadwaj, A. Denny, A. M. Pamadi, S. Christa, and S. Padmavathi, "A Survey of Adversarial Attack and Defence Methods for Deep Learning Models," *Data Science & Exploration in Artificial Intelligence*, pp. 79–87, Feb. 2025.
[17] V. S. Deepthi and S. Vagdevi, "Behaviour Analysis and Detection of Blackhole Attacker Node under Reactive Routing Protocol in MANETs," *Proc. Int. Conf. Networking, Embedded and Wireless Systems (ICNEWS)*, 2018.
[18] M. Tajuddin and A. N. Harigol, "Intrusion Detection of Imbalanced Network Traffic Based on Machine Learning and Deep Learning Tech- nique," *Int. J. Innovative Research in Computer and Communication Engineering*, vol. 10, no. 7, pp. 10, July 2022.