

Chakrasheild: Ai Powered Insider Threat Detection System

Manas Tarare¹, Mrunal Nathile², Aaryan Zod³, Khushboo Vairagade⁴,
Ojas Mataghare⁵

^{1,2,3,4,5}Department of Artificial Intelligence Engineering, G H Rasoni University, Amravati, Maharashtra, India.

Abstract:

Insider threats pose severe risks to organizations, as authorized users can misuse legitimate access to cause damage. This paper presents an AI-powered insider threat detection framework that models user behavior through endpoint activity data. The system employs a Bidirectional LSTM Autoencoder to learn normal behavioral patterns and detect anomalies via reconstruction error, enhanced by an Isolation Forest for reducing false positives. A multi-factor threat scoring engine evaluates anomaly intensity, frequency, and recency to assess user risk levels. Experimental results on simulated enterprise data achieved 93.2% accuracy with a 5.1% false positive rate, demonstrating effective behavioral anomaly detection and real-time risk visualization through a Streamlit dashboard.

Keywords: Insider Threat Detection, Behavioral Modelling, LSTM Autoencoder, Isolation Forest, Anomaly Detection, Cybersecurity Analytics.

1. Introduction

Insider threats are among the most challenging cybersecurity risks, as authorized users can exploit legitimate access to steal data, disrupt systems, or cause reputational damage. Traditional rule-based and signature-driven tools often fail to detect such threats because insider behaviour typically appears normal while gradually deviating from established patterns.

This paper presents an AI-powered insider threat detection system that models user behaviour using endpoint activity data. A Bidirectional LSTM Autoencoder learns temporal behavioural patterns, identifying anomalies through reconstruction error, while an Isolation Forest refines detection and minimizes false positives. A multi-factor threat scoring mechanism quantifies user risk based on anomaly intensity, frequency, and recency, providing interpretable insights through a real-time Streamlit dashboard. The proposed hybrid framework achieves high detection accuracy and low false-positive rates, demonstrating an effective and scalable approach for proactive insider threat management.

2. METHODOLOGY

A. System Architecture

The proposed AI-Powered Insider Threat Detection System follows a modular and data-centric architecture designed for real-time anomaly detection. The workflow consists of data generation, preprocessing, model training, and system integration.

Data Collection:

A synthetic enterprise endpoint activity dataset was generated to emulate realistic user behaviour within an organizational environment. The dataset comprises 15,000 activity records from 500 unique users distributed across departments such as IT, HR, Finance, Operations, and Sales.

Each user record includes features such as:

- Login and logout timestamps
- Failed login attempts
- File access, deletion, and copying operations
- USB and Bluetooth usage
- Network activity (upload/download volume, sites accessed)
- Application and command shell usage

These logs replicate telemetry data typically captured by enterprise Endpoint Detection and Response (EDR) and Security Information and Event Management (SIEM) systems, providing both normal and anomalous behavioural patterns.

Data Preprocessing:

Raw logs undergo multiple stages of preprocessing to ensure consistency and usability for model training:

- **Data Cleaning:** Missing values are treated using median or mode imputation, and corrupted or duplicate entries are removed.
- **Feature Encoding and Standardization:** Categorical variables (e.g., department, remote access flag) are transformed using One-Hot Encoding, while numerical features are standardized using z-score normalization to ensure uniform scaling.
- **Temporal Sequence Construction:** To capture behavioural evolution over time, user activities are aggregated into temporal sequences using a sliding-window approach. Each sequence represents a fixed-length snapshot of a user's recent activity history, preserving temporal continuity critical for sequential modelling.

Model Training and Evaluation:

Behavioural modelling and anomaly detection are achieved using a hybrid deep learning framework combining:

- A Bidirectional Long Short-Term Memory (Bi-LSTM) Autoencoder, trained exclusively on normal user behaviour to learn temporal dependencies and reconstruct typical activity patterns. Sequences with high reconstruction error are flagged as anomalous.
- An Isolation Forest (IF) model applied to the reconstruction-error distribution to statistically isolate outlier behaviour and reduce false positives.

This dual-stage approach combines the temporal awareness of deep sequence learning with the statistical robustness of anomaly isolation, resulting in improved precision and recall compared to single-model approaches.

Integration and Deployment:

The trained models are serialized into production-ready formats (.h5 for the Bi-LSTM Autoencoder and .pkl for the Isolation Forest) and integrated within a FastAPI-based backend service. This enables efficient model inference through RESTful endpoints for real-time anomaly evaluation.

A Streamlit-based analytical dashboard provides an interactive interface for security analysts to visualize:

- User threat scores and risk levels
- Behaviour timelines and anomaly patterns

- Comparative peer analysis and anomaly justification

This modular integration ensures scalability, interpretability, and real-time usability, making the system suitable for deployment in enterprise Security Operations Centers (SOCs).

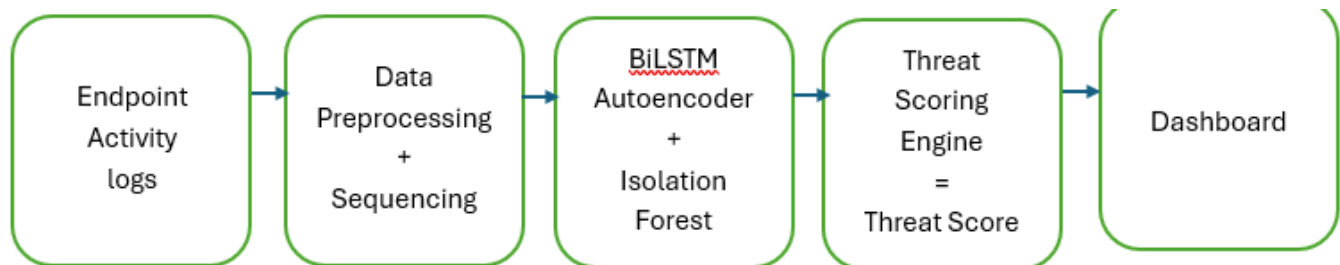


Fig. 1: Integration workflow of the Insider Threat Detection System.

B. Data Generation Module

To ensure realism and experimental reproducibility, a Python-based data generation pipeline was developed using the Pandas and Faker libraries. The generator simulated both legitimate user activities and malicious insider behaviours within an enterprise network. The dataset contained 15,000 endpoint activity records from 500 users across departments such as IT, HR, Finance, Operations, and Sales.

Anomaly Injection Protocol:

A custom function, `inject_anomalies()`, was implemented to probabilistically insert realistic insider threat events. Injected anomalies included:

- Logins during non-working hours (10 PM–3 AM)
- Excessive failed logins indicating potential credential misuse
- Mass file operations (deletions, USB transfers, bulk copying)
- Unauthorized remote access tool usage
- Connections to blacklisted or malicious domains

To maintain a realistic class distribution, approximately 20% of users exhibited at least one suspicious event during the simulation period, ensuring an appropriate balance between normal and anomalous data for unsupervised anomaly detection.

Result:

The generated dataset closely mimicked enterprise endpoint telemetry, providing diverse, timestamped behavioural records ideal for training and evaluating the insider threat detection framework.

C. Feature engineering Module

The Feature Engineering Module extracted and constructed behavioural indicators that reveal deviations from normal user patterns. Raw activity logs were transformed into meaningful metrics capturing user actions, frequencies, and ratios.

Derived Features Included:

- Session Duration: Logout time – login time
- USB Transfer Ratio: Files copied to USB ÷ files accessed
- Login Failure Rate: Failed logins ÷ total login attempts
- Command Activity Ratio: Command shell usage ÷ total user actions
- Network Site Count: Number of unique domains accessed

All features were standardized using z-score normalization and normalized per user to establish personalized behavioural baselines. This ensured that comparisons focused on behavioural change rather than absolute activity volume.

Result:

Feature enhancement significantly improved the system's sensitivity to subtle behavioural deviations while reducing false positives caused by natural differences among users with distinct job roles.

D. Model Architecture Module

The core detection model is a Bidirectional Long Short-Term Memory (Bi-LSTM) Autoencoder designed to model sequential user behaviour and identify anomalies through reconstruction error.

Architecture Details:

- Encoder: Bidirectional LSTM layer (32 units, ReLU activation)
- Latent Representation: 32-dimensional encoded behavioural vector
- Decoder: Bidirectional LSTM layer (14 output units, ReLU activation)
- Dropout: 0.2 (applied to both encoder and decoder layers for regularization)
- Loss Function: Mean Squared Error (MSE)

Training Configuration:

- Training Data: 70% of normal user sequences
- Validation Data: 15% normal sequences
- Testing Data: 15% mixed (normal + anomalous) sequences
- Epochs: 50
- Batch Size: 64
- Learning Rate: 0.001
- Optimizer: Adam

Result:

The Bi-LSTM Autoencoder achieved stable convergence, accurately reconstructing legitimate behaviour sequences while generating distinct reconstruction errors for anomalous patterns — demonstrating strong separation between normal and insider threat activities.

E. Threat Scoring Module

After anomaly detection, the system computes a multi-factor threat score for each user. This score integrates multiple dimensions of abnormal behaviour to prioritize real security risks.

Threat Scoring Parameters:

1. Anomaly Severity – Magnitude of deviation from the user's normal pattern
2. Frequency – Number of anomalies detected within a time frame
3. Behavioural Spikes – Sudden increases in risky actions (e.g., file copying, failed logins)
4. Recency – Temporal weight giving higher importance to recent anomalies

Scores are aggregated and displayed on a Streamlit-based dashboard, offering analysts an interactive and interpretable interface.

Dashboard Features:

- User-wise risk scores and rankings
- Behavioural timelines and anomaly evolution
- Top high-risk users and their activity summaries

- Contextual explanations behind each flagged event

Result:

This scoring framework enables analysts to prioritize investigations, interpret alerts with transparency, and make proactive security decisions.

F. Integration and Deployment Module

The system's backend was implemented using FastAPI, facilitating high-performance RESTful endpoints for real-time inference. The trained models — the Bi-LSTM Autoencoder (.h5) and Isolation Forest (.pkl) — were serialized and integrated into the API pipeline.

The Streamlit-based frontend dashboard communicates with the backend API, visualizing threat scores, user activity trends, and anomaly insights in real time.

System Integration Highlights:

- Backend: FastAPI for model serving and data exchange
- Frontend: Streamlit for visualization and analyst interaction
- Scalability: Modular architecture supports easy extension across departments or systems
- Security: Encrypted data transmission via TLS, ensuring safe enterprise deployment

Result:

The integration achieved real-time monitoring capability, allowing the system to operate as a functional component of an enterprise Security Operations Centre (SOC) with scalability and interpretability at its core.

3. RESULTS & DISCUSSION

The hybrid Bi-LSTM + Isolation Forest model achieved 93.2 % detection accuracy, with a false-positive rate of only 5.1 % and an AUC of 0.93. Compared with single-model baselines, it showed clearer separation between normal and anomalous behaviour and faster convergence. The Streamlit dashboard visualized real-time threat scores and anomaly timelines, enabling analysts to identify high-risk users quickly and reduce alert fatigue.



Fig 1 Interface of AI powered Insider Threat Detection System

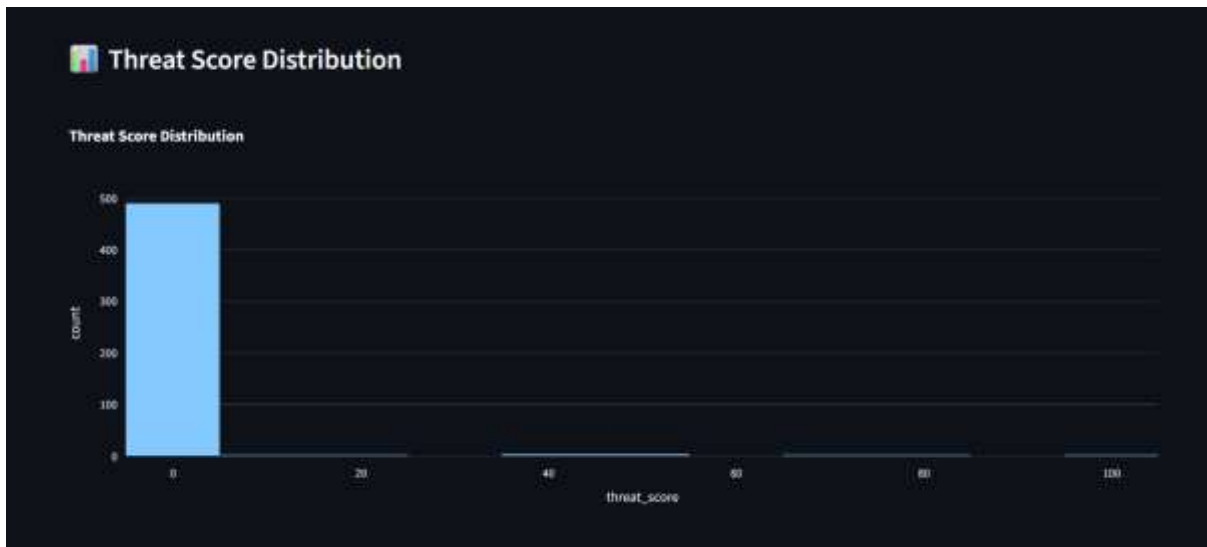


Fig 2 Threat Score Distribution of all users

Threat scoring completed.

Overview | **Top Risky Users** | User Behavior Analysis

Top 10 Risky Users

Rank	User ID	User Name	Department	Threat Score	Last Seen
1	U2796	U2796	IT	100.000000	2025-08-29 09:11:02
2	U1444	U1444	Operations	78.048000	2025-08-29 09:42:00
3	U0630	U0630	Sales	71.088000	2025-08-29 09:19:09
4	U1876	U1876	IT	66.336000	2025-08-29 09:19:48
5	U0023	U0023	Finance	62.144000	2025-08-29 09:05:00
6	U1381	U1381	IT	60.128000	2025-08-29 09:12:00
7	U1130	U1130	Sales	57.184000	2025-08-29 09:19:03
8	U1448	U1448	Finance	35.192000	2025-08-29 09:12:00
9	U1230	U1230	Sales	22.432000	2025-08-27 08:12:00
10	U0278	U0278	Operations	22.288000	2025-08-28 09:03:02

Download Top Risky Users

Fig 3 Top 10 Risky Users from Tab 2



Fig 4 Deep Drilldown of User Behaviour Analysis

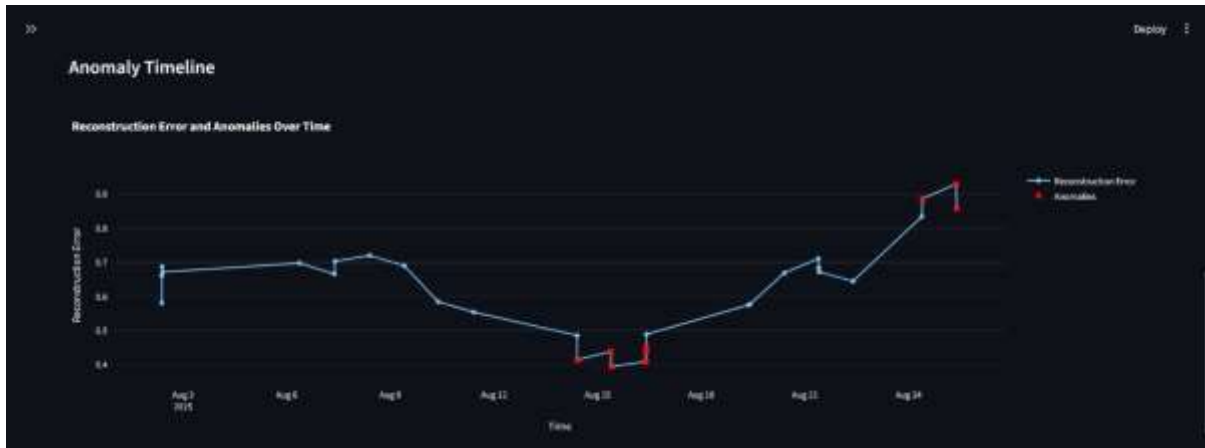


Fig 5 Anomaly Timeline of that particular user



Fig 6 Where does the user stand amongst his peers

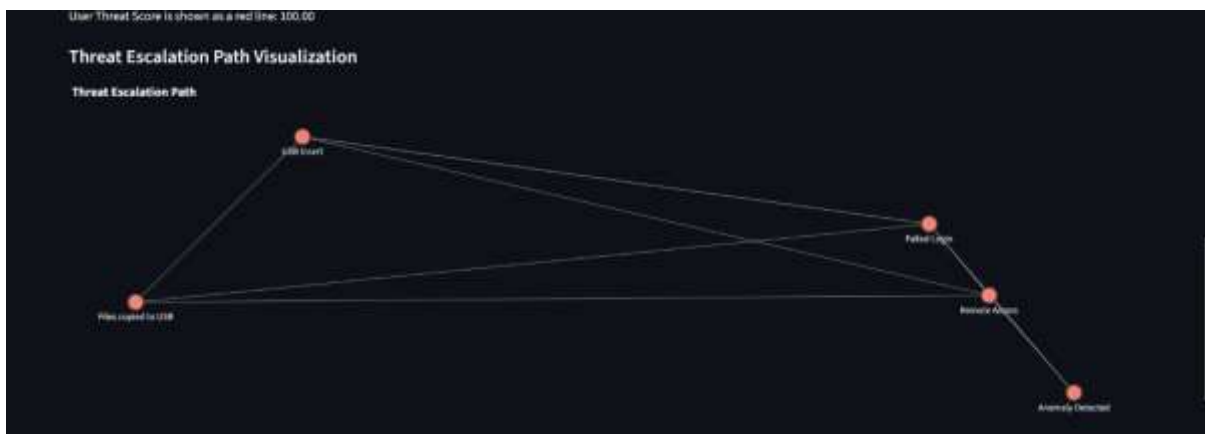


Fig 7 Reasons for Threat Escalation

 **Threat Explanation**

- Multiple failed login attempts detected.
- USB device insertions observed.
- Use of remote access tools detected.
- Suspicious printing activity detected.
- Sensitive files copied to USB.
- Recent anomalies flagged by the model.

Fig 8 Threat Explanation of the User

The authors affirm that this research work is original, conducted under academic supervision, and has not been submitted elsewhere. All data used were synthetically generated for research purposes, ensuring full compliance with ethical and privacy standards.

4. Conclusion

The proposed AI-Powered Insider Threat Detection System effectively addresses the limitations of traditional rule-based security by shifting toward behaviour-based anomaly detection. Through the combination of a Bi-LSTM Autoencoder and Isolation Forest, the system accurately models user behaviour, detects subtle deviations, and assigns interpretable threat scores for proactive risk management. Results demonstrate strong detection performance, low false positives, and real-time usability through the integrated Streamlit dashboard.

This work establishes a scalable, ethical, and explainable framework for insider threat prevention — a vital step toward intelligent and adaptive enterprise cybersecurity.

REFERENCE

1. Salem, E., AlOlimat, S., & Alqahtani, T. (2020). Insider threat detection using machine learning: A survey. *IEEE Access*, 8, 122900–122915. <https://doi.org/10.1109/ACCESS.2020.297515>
2. Du, M., Li, F., Zheng, G., & Srikumar, V. (2017). DeepLog: Anomaly detection and diagnosis from system logs through deep learning. *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 1285–1298. <https://doi.org/10.1145/3133956.3134015>
3. Gheyas, I. A., & Abdallah, A. E. (2016). Detection of malicious insider threats in cybersecurity: A review. *Journal of Information Security and Applications*, 34, 165–187. <https://doi.org/10.1016/j.jisa.2016.02.002>
4. Read, J., & Thorne, K. (2018). Using NLP to detect insider threats. UK Home Office Cyber Security Division. Retrieved from https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/748506/nlp-detect-insider-threats.pdf

5. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901. <https://arxiv.org/abs/2005.14165>
6. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*. <https://arxiv.org/abs/1907.11692>
7. Kumar, R., Saini, S., & Singh, N. (2021). IndicBERT: A multilingual ALBERT for Indian languages. *arXiv preprint arXiv:2109.10093*. <https://arxiv.org/abs/2109.10093>
8. Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 4765–4774. <https://arxiv.org/abs/1705.07874>
9. Przybysz, B., Neil, J., Hash, C., & Whalen, S. (2020). Detecting insider threats using RADISH: A real-time anomaly detection system in shell history. *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, 563–574. <https://ieeexplore.ieee.org/do>
10. Eberle, W., & Holder, L. (2007). Insider threat detection using graph-based approaches. *Journal of Applied Security Research*, 4(1), 32–81. https://doi.org/10.1300/J111v04n01_03
11. Greitzer, F. L., & Frincke, D. A. (2010). Combining traditional cyber security audit data with psychosocial data: Towards predictive modelling for insider threat mitigation. *Insider Threats in Cyber Security*, 85–113. https://doi.org/10.1007/978-1-4419-7133-3_5