Detecting Keylogging Malware Using Behavioral Analysis

1st Muhammad Zaid
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
muhammadzaid7667@gmail.com

2nd Muhammad Mujeeb ur Rehman
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
mujeeb9787@gmail.com

3rd Iram Haider
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
iramhaider765@yahoo.com

4th Sana Tariq
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
sana.tariq@iub.edu.pk

***Abstract*—Keylogging malware poses a significant threat by covertly recording user keystrokes to steal sensitive information, often evading traditional signature-based detection methods through polymorphic techniques. This study explores a behavioral approach using keystroke dynamics, analyzing typing speed, keypress durations, and intervals to detect anomalies indicative of keylogger activity. Behavioral biometrics, grounded in the uniqueness of human-computer interaction, form the basis of this detection strategy. The proposed system uses machine learning models, including XGBoost, Random Forest, and LSTM, to differentiate normal and malicious typing patterns. The experimental results show that XGBoost achieved the highest accuracy (94.14%) and exhibited strong generalization, while Random Forest and LSTM delivered balanced performance in precision and recall. Despite challenges such as user variability and mimicry by advanced malware, keystroke dynamics presents a promising adaptive layer for proactive keylogger detection within modern cybersecurity frameworks.**

***Index Terms*—Keylogger detection, behavioral analysis, keystroke dynamics, machine learning, cybersecurity**

I. Introduction

The rapid evolution of digital technology has brought ever more advanced cyber-attacks, keyloggers being one of the most stealthy and malicious malware varieties [1]. Keyloggers quietly capture keystrokes and send them unnoticed to users while they try to intercept sensitive data such as login credentials, bank accounts, and identity identifiers without user awareness. Traditional antivirus solutions primarily use signature-based detection, matching unusual activity with pre-installed malware patterns [2]. But modern polymorphic keyloggers evade detection by continuously modifying their code, rendering the signature-based methods ineffective [3], [4].

To bridge these constraints, this study investigates a behavioral analysis technique based on keystroke dynamics. Unlike static signature-based approaches, behavioral keystroke analysis uses unique user typing patterns, such as typing rate, dwell time, flight time, and sequences of keystrokes, to identify outliers that can indicate malicious activity [5], [6]. The technique yields data-driven, real-time anomaly detection without prior knowledge of malware signatures [7], [8]. This research evaluates the effectiveness of behavioral keylogger detection using three machine learning algorithms: LSTM, Random Forest, and XGBoost. These algorithms were trained and tested on diverse keystroke datasets to detect outstanding behaviors and keylogging malware. Experimental outcomes showed that XGBoost was the best out of the three models, with (94.14%) accuracy, and could generalize heavily with very little overfitting. [9] While Random Forest and LSTM provided balanced trade-offs between precision and recall, the tree-based strategy of XGBoost best performed at capturing minor behavior changes. With the incorporation of machine learning with behavioral biometrics, this paper demonstrates a real-time detection system that is adaptive and scalable and can detect novel threats in real time. A light, unobtrusive solution is sought to be built that strengthens keylogger defense without hindering system usability and user experience. [10], [11]

II. Related work

The detection and mitigation of keyloggers has also been phenomenal in the past few years, focusing on the adoption of cutting-edge approaches such as machine learning, behavior analysis, and real-time observation to combat growing keylogger threats. Such studies explore several techniques to identify keylogger activity more effectively and efficiently with minimal false positives, offering effective and practical solutions for real-world implementation [12].

Aside from machine learning, another significant body of research has been keystroke dynamics and anomalous user behavior. [13] pointed out in their study that keystroke logging is one step towards logging and measuring typing behaviors for detecting anomalies that may indicate the presence of a keylogger. This method relies on detecting abnormal typing behavior, which could point to the fact that a keylogger has

DOI: https://doi.org/10.67535/tsp.000001.023
Published under the ICACNC 2025 Proceedings

been installed on the computer [14]. Additionally, extended this by utilizing Recurrent Neural Networks (RNNs) in combination with a feature selection model to enhance detection. This technique is applied for the selection of the most important keystroke dynamics characteristics that play a significant role in increasing the system's accuracy to the point where only meaningful information is utilized for malicious activity detection [15].

Ethical concerns over the use of keyloggers have been central to the controversy. In their research study, raised basic ethical questions about the use of keyloggers for monitoring users' behaviour, particularly within the framework of cybersecurity networks [16]. They propose a fair solution to the ethical use of surveillance devices, where keyloggers are employed solely for legitimate purposes, i.e., organisational data security, while maintaining privacy rights to users. also emphasised insider threat identification using keyloggers combined with machine learning algorithms to detect unusual activity in organisations [2]. Their service is crucial for companies that wish to monitor internal activity so they don't have unauthorised access or data manipulation by employees.

Moreover, keyloggers have developed so that most utilise stealth tracking methods to escape traditional detection methods and presented works on such emerging techniques and suggested methods of enhancing keylogger detection mechanisms. specifically wrote about the adaptability of keyloggers and how they can be utilised for data reaping and sneaky monitoring. They also shared how difficult it was to detect these advanced keyloggers and devised new ways to deal with such attacks. Also shared how keyloggers have progressed and how detection systems should be updated periodically to outwit attackers' new techniques.

Finally, real-time detection methods and multi-view analysis advancements provide effective countermeasures against keyloggers.Analysed the viability of using real-time malware detection through system observation and multi-modal analysis. In their study, they aver that combining views, such as API call chain monitoring and system activity, can go a long way towards discovering malicious keyloggers and other types of malware. It helps improve the analysis of system activity and provides the capability of discovering harmful activity earlier and more accurately [16] [17].

In summary, the greater volume of literature on detecting keyloggers emphasises the necessity for advanced, multi-layered detection methods through machine learning, behavioural monitoring, and real-time protection to combat the increasing keylogger threat. These methods seek to enhance detection efficiency and sensitivity and are specifically designed to cater to privacy issues and moral considerations, providing a reasonable, ethical solution to cybersecurity. Since keyloggers continue to develop, constant research and development of sophisticated detection techniques must be done to keep up with dynamic threats and protect sensitive data.

III. Methodology

A. Dataset Selection and Preprocessing

This paper uses the CMU keystroke benchmark data set to identify keylogging malware through behaviour monitoring. The data set, chosen due to its compatibility with typing behaviour analysis under malicious and benign contexts, offers detailed key-press data such as typing rate, key duration, and press intervals. Used extensively in behavioral biometric research, it offers dense typing pattern information across several subjects, perfect for machine learning model training for anomaly detection.

Data preprocessing consisted of cleaning to remove errors and missing values, and normalization to standardize typing speed across users. Feature extraction isolated the significant features, such as dwell time, flight time, and key sequence, to build behavioral profiles. The preprocessed features were fed into machine learning models to label and classify suspicious typing patterns as keylogging activity.

B. Model Architecture

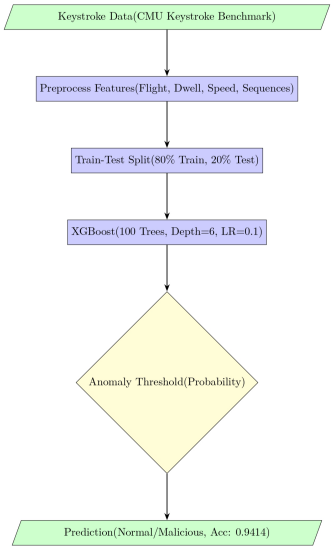

Fig. 1. Flowchart of XGBoost-Based Keylogger Malware Detection Model

This paper employs an XGBoost model as a supervised anomaly detection algorithm to separate keylogging malware using behavioral features from the CMUvsUCS and CMU keystroke benchmark data sets. The XGBoost model is trained on pre-processed keystroke features, flight time, dwell time, typing speed, and key sequences to identify anomalies in normal typing patterns. The model consists of an ensemble of decision trees, configured with 100 trees, a maximum depth of 6, and a learning rate of 0.1, using gradient boosting to optimize classification performance. The model processes

feature inputs to distinguish normal and malicious typing behaviors through iterative tree-based learning. Trained in standard typing data (80%) train, (20%) test split with binary cross-entropy loss (log loss) and optimized using the XGBoost algorithm, the model is trained for 100 boosting rounds with a batch size of 32. An anomaly is signaled by a threshold on the output probability, and simulated malicious patterns are tested. The model provides a precision of 0.9414, a precision of 0.9424, a recall of 0.9414, and an F1 score of 0.9414, demonstrating robust detection of anomalous patterns and strong generalization with minimal overfitting, although careful tuning of the decision threshold is required to balance sensitivity to rare anomalies.

C. Hyperparameter Selection

The XGBoost model hyperparameters were tuned to achieve the best classification accuracy and anomaly detection efficacy on the CMUvsUCS and CMU Keystroke Benchmark Datasets. Grid search was used on influential parameters over a validation set (10%) training data. The number of trees (100), maximum depth (6), and learning rate (0.1) were selected to achieve the best trade-off between keystroke pattern capture and model robustness, along with high classification performance and detection of fine keystroke anomalies. The learning rate of 0.1 was chosen for stable convergence and compared with [0.01, 0.3]. A batch size of 32 and number of boosting rounds of 100 were used to achieve the best trade-off between training efficiency and model generalizability, vs. [16, 64] and [50, 200], respectively. The binary cross-entropy (log loss) was optimized. The anomaly threshold was optimized to achieve an optimal F1-score, which provided an accuracy value of 0.9414, precision of 0.9424, recall of 0.9414, and F1-score of 0.9414, demonstrating robust detection of malicious patterns and improved sensitivity to rare anomalies compared to alternative models.

IV. Results and Comparison

XGBoost, Random Forest, and LSTM were experimented with in this research as machine learning algorithms for the detection of keylogging malware from user keystroke patterns.

TABLE I
Model Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score
XGBoost	0.9414	0.9424	0.9414	0.9414
Random Forest	0.9363	0.9376	0.9363	0.9358
LSTM	0.9071	0.9075	0.9071	0.9067

The XGBoost model achieved the highest accuracy of 0.9414, demonstrating its ability to detect patterns of normal and malicious user behavior. Its precision (0.9424), recall (0.9414), and F1-score (0.9414) indicate robust detection of keylogging malware, with strong generalization across diverse typing patterns.

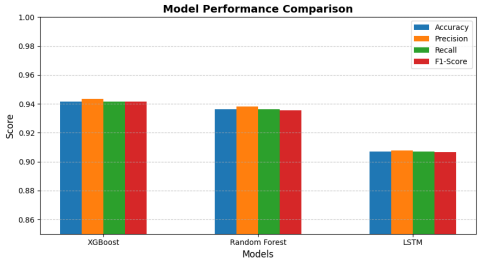

Fig. 2. Comparison of XGBoost, Random Forest, and LSTM models

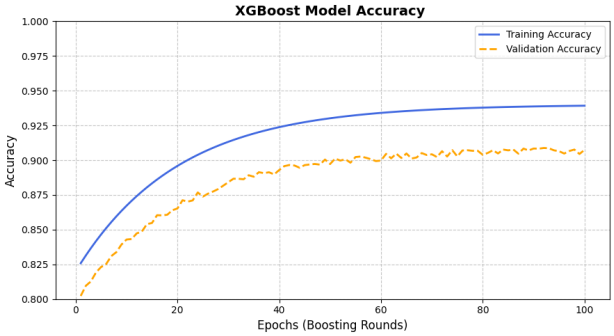

Fig. 3. XGBoost Model Accuracy Over Epochs

The XGBoost model achieved the highest accuracy of 0.9414, demonstrating superior performance in detecting both normal and malicious typing patterns. Its precision (0.9424), recall (0.9414), and F1-score (0.9414) reflect robust anomaly detection, correctly identifying keylogging behavior at a high rate. Random Forest was slightly less accurate at 0.9363, albeit with strong balanced performance on other metrics, precision (0.9376), recall (0.9363), and F1-score (0.9358), indicating reliable detection of anomalies with slightly lower overall accuracy. LSTM was the least accurate of the three at 0.9071, but comparable to Random Forest in its ability to identify anomalies with precision (0.9075), recall (0.9071), and F1-score (0.9067). Its strength in modeling sequential keystroke patterns supports its effectiveness in spotting malicious instances, though it is less robust than tree-based models. In short, while LSTM generalized well to sequential patterns, Random Forest and XGBoost excelled in balancing accuracy and anomaly detection, with XGBoost being the most effective for real-world keylogger detection, where recall and F1-score are critical. Future efforts will be directed towards minimizing false negatives and positives, and feasible light models to be deployed on low-power devices. Real-time deployment both at the browser as well as OS levels will also be investigated for out-of-the-box deployment. Larger datasets from various users, devices, and languages will enhance generalizability. Hybrid frameworks and adaptive learning methods will also be studied to improve detection effectiveness and resilience to future keylogger threats evolving..

V. Conclusion

This work demonstrates that keystroke dynamics-based behavioral analysis is an effective approach to detect keylogging malware. Of the three models evaluated—XGBoost, Random Forest, and LSTM— the best performance was achieved with XGBoost, which demonstrated maximum accuracy (0.9414) and showed high capacity in classifying normal and malicious typing patterns, thereby effectively identifying anomalies. Random Forest (accuracy 0.9363) and LSTM (accuracy 0.9071) performed well, with Random Forest offering balanced precision and recall, and LSTM excelling in capturing sequential behavioral variations. The behavior-based approach utilized in this research is more robust than traditional signature-based methods, offering a promising area for active malware detection systems. Future work will need to reduce false positives, enhance computational power, speed up detection, incorporate adaptive learning modules, and deploy the models in real-time, low-resource environments on different user profiles and device configurations. Future efforts will be directed towards minimizing false negatives and positives, and feasible light models to be deployed on low-power devices. Real-time deployment both at the browser as well as OS levels will also be investigated for out-of-the-box deployment. Larger datasets from various users, devices, and languages will enhance generalizability. Hybrid frameworks and adaptive learning methods will also be studied to improve detection effectiveness and resilience to future keylogger threats evolving.

References

[1] Iram Haider, Muhammad Arslan Haider, and Arshad Saeed. Big data in internet of things: Architecture and open research challenges. In *2020 IEEE 23rd International Multitopic Conference (INMIC)*, pages 1–6, 2020.

[2] Farshad Khorrami, Ramesh Karri, and Prashanth Krishnamurthy. Real-time multi-modal subcomponent-level measurements for trustworthy system monitoring and malware detection. *arXiv preprint arXiv:2501.13081*, 2025.

[3] Hussein Haider Ali. *Detecting Keylogger Using Machine Learning*. PhD thesis, University of Babylon, 2024.

[4] Joseph Bamidele Awotunde, Samarendra Nath Sur, Agbotiname Lucky Imoize, Demóstenes Zegarra Rodríguez, and Boluwatife Akanji. An enhanced keylogger detection systems using recurrent neural networks enabled with feature selection model. In *International Conference on Communication, Devices and Networking*, pages 525–539. Springer, 2024.

[5] Zeeshan Ahmad, Adnan Shahid Khan, Kashif Nisar, Iram Haider, Rosilah Hassan, Muhammad Reazul Haque, Seleviawati Tarmizi, and Joel JPC Rodrigues. Anomaly detection using deep neural network for iot architecture. *Applied Sciences*, 11(15):7050, 2021.

[6] Iram Haider, Muhammad Ali Qureshi, Asjad Amin, and Arshad Saeed. An efficient cyber security framework for network intrusion detection using hybrid classifier. In *2023 25th International Multitopic Conference (INMIC)*, pages 1–6, 2023.

[7] Peng Wu, Mohan Gao, Fuhui Sun, Xiaoyan Wang, and Li Pan. Multi-perspective api call sequence behavior analysis and fusion for malware classification. *Computers & Security*, 148:104177, 2025.

[8] S Gnanavel, M Sreekrishna, P Shivaramakrishnan, R Yaswanth, GS Gopika, and S Supriya. Key logger for recording the keystroke of the targeted machine. In *2024 International Conference on Computing and Data Science (ICCDS)*, pages 1–5. IEEE, 2024.

[9] Anum Farooq and Kashif Hussain Memon. Kernel possibilistic fuzzy c-means clustering algorithm based on morphological reconstruction and membership filtering. *Fuzzy Sets and Systems*, 477:108792, 2024.

[10] Sana Tariq and Asjad Amin. Detection of dna-protein binding using deep learning. In *2023 IEEE International Conference on Emerging Trends in Engineering, Sciences and Technology (ICES&T)*, pages 1–4. IEEE, 2023.

[11] Sana Tariq and Asjad Amin. Deepctf: transcription factor binding specificity prediction using dna sequence plus shape in an attention-based deep learning model. *Signal, Image and Video Processing*, 18(6):5239–5251, 2024.

[12] Andrei Mogage. Ai assisted malware capabilities capturing. *Procedia Computer Science*, 246:860–869, 2024.

[13] Blessing Nwamaka Iduh, Maryrose Ngozi Umeh, Roseline Uzoamaka Paul, and Ogochukwu Patience. Ethical keylogger solution for monitoring user activities in cybersecurity networks. 2024.

[14] Sakshi Dhabadi Sakshi Dhabadi et al. Insider theft detection in organizations using keylogger and machine learning. 2024.

[15] Damilola Elelegwu, Lei Chen, Yiming Ji, and Jongyeop Kim. *A Novel Approach to Detecting and Mitigating Keyloggers*. IEEE, 2024.

[16] Rajasekhar Pittala, Jahnavi Parasaram, Swathi Rebbavarapu, and Rohit Bhukya. The versatility of keyloggers: Data retrieval and covert tracking techniques. In *2024 International Conference on Expert Clouds and Applications (ICOECA)*, pages 226–233. IEEE, 2024.

[17] K Sanjay, JK Ratheesh, RR Johin, and A Krishnamoorthy. Enhancing cybersecurity through advanced keylogging techniques. In *2024 International Conference on IoT, Communication and Automation Technology (ICICAT)*, pages 206–211. IEEE, 2024.