Behavior Based Continuous Authentication and Remote Security Management System

Dr. P. Thamarai[1], Kosana Manohar[2], Kuridi Dwaraka Sai Ganesh[3],
Kommisetti Sai Surya Manikanta[4], Kuracha Sri Satya Murali Raghava[5]
[1]*Assisstant Professor Bharath Institute of Science and Technology Chennai, India*
[2,3,4,5]*CSE dept, Bharath Institute of Science and Technology Chennai, India*

***Abstract*—The Behavioral Biometrics Security Control System is a comprehensive security framework designed to enhance intrusion detection and user authentication through continuous behavioral analysis. Unlike traditional authentication methods that rely on passwords or PINs, which can be stolen or shared, this system introduces an additional layer of security by monitoring how a user interacts with a device. By tracking mouse movements, clicks, and scrolling patterns, it builds a unique behavioral profile that reflects natural interaction style, making it difficult for unauthorized users to replicate. At the core of the system is a One-Class Support Vector Machine model suited for anomaly detection. It is trained only on legitimate user data to learn normal behavior patterns. During real-time usage, incoming mouse activity is continuously analyzed and compared with this profile. If deviations exceed a defined threshold, the behavior is classified as suspicious, enabling detection even after login and ensuring continuous authentication. When suspicious activity is identified, the system responds by capturing webcam images with timestamps and generating alerts. These records are displayed on a web dashboard, providing visual evidence and improving accountability and traceability. The framework uses a multi-component architecture integrating a Python backend, Linux service, and web interface. The backend manages data processing, feature extraction, and model execution, while the Linux service ensures uninterrupted background monitoring. The dashboard centralizes system status and intrusion events. An Android application extends remote access, allowing users to view system metrics and detected anomalies. It also enables remote actions such as locking, logging out, or shutting down the system. Overall, the system delivers adaptive, continuous, and user-friendly security. This approach strengthens protection against unauthorized access while maintaining simplicity and efficiency. It is suitable for personal computers, enterprise systems, and secure environments where continuous verification and rapid response are essential for modern cybersecurity needs and future authentication systems.**

I. INTRODUCTION

In the digital age, the exponential growth of online services and the increasing reliance on web-based platforms have made user authentication one of the mast critical aspects of cybersecurity. Traditional authentication mechanisms such as passwords, PINs, or biometric scans are widely used to verify user identities.

However, these methods provide only one-time authentication that is, verification occurs at the initial point of access. Once authenticated, the system implicitly assumes that the same legitimate user continues the session, which creates opportunities for session hijacking, credential theft, and unauthorized access. To address these vulnerabilities, researchers have shifted their focus toward continuous authentication mechanisms, which monitor and verify user identity throughout the session, providing a more dynamic and adaptive security layer. Among various behavioral biometrics, cursor dynamics and scrolling behavior have emerged as promising modalities for continuous authentication. Cursor movements, mouse clicks, and scrolling patterns are unique to individuals and can reflect subtle aspects of their motor control, hand-eye coordination, and Interaction style. These behavioral traits are difficult to mimic, making them suitable for passive and non-Intrusive authentication systems. Unlike traditional methods that interrupt user activity, behavioral-based continuous authentication operates in the background, ensuring both security and usability without requiring explicit user actions.

The concept of cursor dynamics involves analyzing a user's mouse movement trajectories, speed, acceleration, pause durations, click frequency, and scrolling intensity. Every user exhibits distinctive movement rhythms while navigating through web pages some users move their cursors in straight, fast paths, while others exhibit more curved, hesitant patterns. Similarly, scrolling behavior, including scroll velocity, scroll direction changes, and scroll intervals, reflects consistent individual tendencies that can serve as behavioral signatures. When combined, these two modalities provide a continuous data stream that can be used to model and verify user identity in real time. Traditional authentication systems rely heavily on static data, which makes them vulnerable to various cyber threats such as phishing, brute force attacks, and stolen credentials. In contrast, continuous behavioral authentication systems function as a secondary defense layer, identifying anomalies that may occur during a session.

For example, if a legitimate user logs in but an attacker takes control of the device later, the system can detect the deviation in cursor movement or scrolling behavior and trigger an automatic logout or re-authentication request. This adaptive monitoring ensures ongoing verification without disrupting user experience, thereby balancing security and convenience two essential yet often conflicting requirements in modern authentication systems.
Furthermore, cursor-based behavioral authentication can be seamlessly integrated into existing web applications without requiring specialized hardware. It only requires software-based tracking mechanisms implemented through JavaScript or similar browser technologies. The collected behavioral data can then be processed using machine learning algorithms such as Random Forest, Support Vector Machines (SVM), or Neural Networks to classify users based on their behavioral profiles. These algorithms learn from labeled behavioral data, identifying subtle differences between users and continuously updating the model for improved accuracy and adaptability.

In recent years, several research efforts have demonstrated that behavioral biometrics can achieve high accuracy in identifying users over time. However, many of these studies have focused on keystroke dynamics or touch-based gestures, primarily in mobile environments. The exploration of cursor and scrolling dynamics on desktop or web-based systems remains relatively less explored, despite its potential for providing continuous authentication on a wide range of web services such as banking portals, e-commerce sites, and corporate dashboards. Given the widespread use of the mouse and touchpad as primary input devices, cursor- based authentication presents a scalable and cost- effective approach to strengthening online security. The proposed project, “Continuous Behavioral Authentication through Cursor Dynamics and Scrolling Behavior,” aims to develop a system that monitors a user’s cursor and scrolling patterns during active sessions and leverages machine learning models to authenticate the user continuously. The primary objectives of the project include
(1) designing a data collection module to capture real-time behavioral data,
(2) extracting meaningful features from raw cursor and scrolling logs,
(3) training and validating classification models for identity verification, and
(4) evaluating system performance in terms of accuracy, reliability, and responsiveness.
The outcome of this project is expected to demonstrate the feasibility and effectiveness of cursor-based behavioral authentication as a supplementary or alternative approach to conventional login systems. In summary, continuous behavioral authentication represents a paradigm shift in digital identity verification from static, one-time checks to dynamic, ongoing monitoring.

By leveraging cursor movement patterns and scrolling behaviors, this project seeks to build an unobtrusive, intelligent, and user-friendly authentication framework that enhances session security while preserving user comfort. Such an approach has the potential to revolutionize the way digital systems perceive and authenticate human users, contributing significantly to the future of cybersecurity, human- computer interaction, and behavioral analytics.

II. LITERATURE SURVEY

The evolution of authentication systems has progressed from traditional static methods to dynamic and intelligent behavioral models aimed at enhancing both security and usability. Conventional authentication approaches, including password-based, token-based, and biometric verification systems, have long been the standard for verifying user identity. However, these methods suffer from several critical drawbacks such as vulnerability to phishing, password theft, and replay attacks, as well as their inability to ensure user legitimacy after the initial login. Once authenticated, the system assumes the user remains the same throughout the session, leaving opportunities for session hijacking and insider threats. To overcome these limitations, researchers have increasingly focused on continuous authentication systems that verify user identity throughout the active session based on behavioral characteristics. Among various behavioral biometrics such as keystroke dynamics, touch gestures, and gait recognition, cursor dynamics and scrolling behavior have recently gained attention for their non-intrusive and universally applicable nature. Behavioral biometrics rely on patterns in how users interact with devices, capturing unique motor control and cognitive characteristics that are difficult to replicate. Early studies by Monrose and Rubin (1997) on keystroke dynamics established the concept of behavior-based authentication by analyzing users' typing rhythms. However, keystroke-based methods are limited to typing-intensive tasks and unsuitable for general browsing environments, which led to the exploration of mouse movement and cursor-based authentication as a more flexible alternative. Ahmed and Traore (2007) were among the first to propose mouse dynamics for continuous authentication, analyzing features such as movement speed, click frequency, and trajectory curvature to distinguish between users. Their findings demonstrated that even simple mouse metrics could effectively differentiate legitimate users from impostors. Subsequent research by Feher et al. (2012) expanded upon this work by incorporating acceleration patterns and movement direction changes, resulting in improved classification performance. Recent developments have applied machine learning algorithms like Support Vector Machines (SVM), Random Forests, and Neural Networks to enhance accuracy and adaptability. Mahmoud et al. (2020) further employed deep learning architectures to model non-linear movement behaviors, achieving greater resilience to behavioral variations over time.

While cursor dynamics have been widely investigated, scrolling behavior has emerged as an additional behavioral feature that captures unique aspects of user interaction, such as reading pace, attention focus, and navigation style. Early evidence from Epp et al. (2011) suggested that scrolling behavior could indirectly reflect user engagement and serve as a subtle biometric indicator. Later studies, such as those by Revett (2016) and Gupta and Banerjee (2021), demonstrated that combining scroll-based metrics with cursor dynamics improved authentication accuracy and system robustness. Features such as scroll velocity, direction reversal, and time intervals between scrolls provided valuable indicators of user identity. The fusion of cursor and scrolling behavior thus creates a more holistic representation of user interaction patterns. In terms of computational models, continuous authentication systems commonly employ machine learning techniques that operate in two phases training and monitoring. During training, behavioral data collected from legitimate users are used to create individual behavioral profiles, while during monitoring, real-time data are compared against these profiles to determine authenticity. Algorithms such as Random Forest and SVM have been favored for their ability to handle non- linear and noisy behavioral data, while recent studies have explored deep neural networks and recurrent neural networks (RNNs) to capture temporal dependencies in sequential behavioral events. Research by Pusara and Brodley (2004) showed that decision tree-based classifiers could achieve over 90% accuracy in identifying impostor sessions, whereas Saxena et al. (2020) improved detection rates through deep learning-based temporal models.

Despite these advancements, several research gaps persist in the field. Many existing studies are based on small or controlled datasets, limiting the generalizability of their findings to real-world web applications. Additionally, cursor dynamics and scrolling behavior have often been examined

independently, without integrating them into a unified continuous authentication framework. Real-time adaptability, which allows models to adjust to gradual behavioral drift, remains another significant challenge. Moreover, computational efficiency and privacy-preserving data collection are vital considerations for deploying such systems at scale. Addressing these gaps, the present study proposes a hybrid behavioral authentication system that leverages both cursor dynamics and scrolling behavior to continuously monitor user identity throughout a session. By utilizing machine learning models to learn and verify user-specific behavioral signatures, the system aims to enhance security while maintaining a seamless and non- intrusive user experience. This literature review thus underscores the growing importance of behavioral biometrics in the context of modern cybersecurity and highlights the need for advanced continuous authentication systems capable of real-time, adaptive, and user-friendly operation across diverse digital platforms.

III. PROBLEM STATEMENT

In today's digital ecosystem, where users frequently access sensitive online platforms such as banking portals, corporate dashboards, and cloud-based applications, ensuring secure and continuous user authentication has become a major challenge. Traditional authentication mechanisms like passwords, PINs, and even biometric methods such as fingerprints or facial recognition provide only static, one-time verification at the time of login. Once the authentication is completed, the system assumes that the same legitimate user remains active throughout the session. However, this assumption exposes systems to serious security vulnerabilities such as session hijacking, credential theft, shoulder surfing, and unauthorized access after the initial login. Attackers who gain access to a system post-login can exploit this static authentication model to perform malicious activities without detection, as there is no mechanism to continuously verify the user's identity. This highlights a critical need for continuous authentication mechanisms that can validate user legitimacy throughout the duration of their interaction. Current research in behavioral biometrics has demonstrated that user interactions such as mouse movement, cursor navigation, and scrolling patterns are inherently unique to individuals and can serve as reliable indicators of identity. These behaviors are influenced by a combination of psychological, physiological, and motor factors, making them difficult for impostors to imitate. Despite their potential, most existing behavioral authentication systems either focus solely on keystroke dynamics or cursor movement without incorporating complementary behavioral features like scrolling behavior, which can provide additional discriminatory power. Moreover, many existing solutions lack real-time adaptability and struggle to maintain accuracy under varying user conditions or device types.

Therefore, the central problem addressed in this project is the development of a continuous, non-intrusive, and reliable authentication system that utilizes cursor dynamics and scrolling behavior to verify user identity throughout an active session. The proposed system aims to monitor user interaction patterns passively, extract relevant behavioral features, and employ machine learning algorithms to distinguish between legitimate users and potential impostors in real time. By continuously analyzing movement trajectories, speed variations, click patterns, and scrolling tendencies, the system can detect anomalies indicating unauthorized access. The goal is to design a lightweight, privacy- preserving, and adaptive solution that enhances security without disrupting the normal user experience.

In summary, the problem lies in the absence of a robust, real-time behavioral authentication model that integrates both cursor dynamics and scrolling behavior for continuous verification. Addressing this problem will not only strengthen session-level security but also reduce the dependency on intrusive re-authentication methods, thereby balancing usability and security in modern web-based environments

IV. EXISTING SYSTEM

The existing behavioral authentication systems primarily focus on differentiating between multiple users based on their interaction patterns, such as cursor dynamics, keystrokes, or touch gestures. However, these approaches face several practical and

technical limitations when applied to real-world continuous authentication scenarios. Most existing systems exhibit a multi-user dependency, requiring behavioral data from a large group of users typically around 30 to 40 participants for effective model training and evaluation. This dependency makes data collection more time-consuming and impractical, especially in cases where the objective is to monitor and authenticate only a single authorized user. Additionally, these systems often rely on multi-class classification models such as Random Forest, K-Nearest Neighbors (KNN), Convolutional Neural Networks (CNN), or Long Short-Term Memory (LSTM) networks. While these models can classify multiple user behaviors with reasonable accuracy, they are computationally intensive and difficult to maintain compared to simpler one-class methods.

Furthermore, most existing studies do not utilize One-Class Support Vector Machines (OC-SVM), which are particularly effective in situations where only genuine user data is available for training. Instead of learning to distinguish among many users, one-class models learn to detect deviations from the genuine user's normal behavior, making them more suitable for continuous authentication tasks. The lack of such one-class approaches reduces the efficiency of current systems in detecting impostors under limited data conditions. Another significant drawback of the existing systems is their high data and hardware requirements. Many deep learning-based models demand large, balanced datasets and powerful computational resources for both training and inference. This restricts their deployment in lightweight, real-time, or resource-constrained environments, such as standard web browsers or low-power devices.

Moreover, the focus of most previous work has been on multi-user classification identifying which user from a group is interacting with the system rather than continuously verifying whether the ongoing user remains the same as the authorized one. This limits the applicability of such systems in single-user security scenarios, where the key objective is to detect any deviation from the legitimate user's normal interaction behavior. As a result, existing systems, although theoretically effective for user identification, often fail to provide practical, real-time, and adaptive authentication for individual user monitoring.

V. PROPOSED SYSTEM

The proposed system introduces an intelligent and continuous authentication framework that leverages cursor dynamics and scrolling behavior to verify user identity throughout an active session. Unlike conventional one-time login systems, this approach continuously monitors and analyzes user interaction patterns to ensure that the active user remains the legitimate one. The system begins by capturing detailed behavioral data from user interactions, including parameters such as cursor movement speed, acceleration, click frequency, movement direction, and scrolling attributes like velocity and direction changes. These features are continuously collected in the background as the user interacts with the interface, ensuring a non- intrusive and seamless authentication experience. The captured behavioral data undergoes preprocessing techniques such as normalization, noise reduction, and outlier filtering to eliminate inconsistencies caused by random movement spikes, hardware sensitivity, or temporary user distractions. This preprocessing step ensures that the input data is both reliable and consistent before being used for model training or verification. The refined data is then passed through machine learning algorithms, which are trained to construct a unique behavioral profile for each user. These models are designed to recognize the natural rhythm and interaction style of a user, enabling them to differentiate legitimate sessions from potential impostors. The system periodically updates the trained model to accommodate natural variations in user behavior over time, thereby maintaining accuracy and adaptability without requiring manual reconfiguration.

During active sessions, the system performs real-time monitoring and comparison between the user's current interaction data and their stored behavioral profile. If the ongoing behavioral pattern aligns closely with the legitimate user's profile, the system allows the session to continue uninterrupted. However, if significant deviations are detected suggesting that an unauthorized person might be using the system the framework automatically

initiates a re-authentication process, prompting the user to verify their identity through secondary means (such as password or OTP). This proactive mechanism helps in preventing unauthorized access even after the initial login.

By combining continuous monitoring with machine learning-driven behavioral profiling, the proposed system provides a non-intrusive, adaptive, and secure authentication method that effectively bridges the gap between usability and cybersecurity. It eliminates the dependence on static credentials and instead relies on the uniqueness of human behavior as a dynamic biometric signature.

One-Class Support Vector Machine (SVM)

The One-Class Support Vector Machine (SVM) is an unsupervised anomaly detection algorithm that is widely used for identifying patterns that differ from normal behavior. Unlike traditional multi-class or binary classifiers that require both positive (genuine) and negative (impostor) samples, the One-Class SVM is trained exclusively on data from a single class typically the genuine user. It then learns a decision boundary that encloses the majority of the data points belonging to this class in a high-dimensional feature space. During testing or real-time operation, any new data point that lies outside this learned boundary is considered an anomaly, indicating possible unauthorized or abnormal behavior.

In the context of continuous behavioral authentication, the One-Class SVM is particularly effective because it can accurately model the unique behavioral characteristics of an individual user based on their cursor movement, scrolling patterns, click frequency, and speed dynamics. The model learns what constitutes “normal” behavior for a specific user and can detect deviations without needing impostor data, which is often unavailable or difficult to collect. The algorithm uses a kernel function commonly the Radial Basis Function (RBF) to transform the input feature space into a higher-dimensional space where a clear separation can be achieved. This enables the SVM to capture complex, non-linear behavioral relationships between different movement and scrolling features. Mathematically, the One-Class SVM works by finding a hyperplane that best separates the data from the origin in the transformed feature space. The optimization goal is to maximize the margin around this hyperplane while minimizing the number of misclassified samples.

A small percentage of data points, known as support vectors, lie near the boundary and define the decision surface. When a new data sample (such as a live cursor movement pattern) is introduced, the model evaluates whether it falls within this boundary (normal behavior) or outside (anomaly).The advantages of using One-Class SVM in this project include its ability to function effectively with only genuine user data, its robustness to noise, and its adaptability to new behavioral trends through periodic retraining. Since it does not rely on attacker data, it avoids overfitting to specific impostor patterns and generalizes well to unseen deviations. Moreover, its computational efficiency makes it suitable for real-time applications like continuous behavioral authentication systems, where low latency is essential.

In summary, the One-Class SVM provides an ideal framework for detecting deviations in user behavior in real time, enabling continuous verification during active sessions. Its capability to model complex behavioral traits with minimal data makes it a powerful component in achieving non-intrusive, adaptive, and secure authentication for modern computing environments.

VI. SYSTEM ARCHITECTURE

A. User Interface (UI)

The User Interface (UI) provides an interactive and user-friendly medium through which authentication results and system messages are communicated. It displays real-time updates about whether the session is secure or requires re-authentication, ensuring the user remains informed without any disruption. The interface can be implemented as a simple console-based display or a web dashboard using frameworks such as Streamlit or Flask. It may also visualize behavioral metrics like cursor movement, scroll patterns, and anomaly detections, giving a clear picture of the authentication process. Overall, the UI acts as the system’s communication layer between the backend processes and the end-user.

B. Backend

The Backend serves as the central processing unit of the entire system. It manages all operations, including data flow, user interaction, and integration between the modules. The primary control script, typically named main.py, initializes the continuous monitoring process and handles real-time authentication tasks. The backend also includes a data-handling script, such as behavior_functions.py, which is responsible for data capture, filtering, normalization, and preparation before model training or prediction. Libraries like Pandas, NumPy, and Scikit-learn are used extensively for preprocessing, numerical computation, and model implementation. This modular backend ensures scalability, maintainability, and efficient execution during both training and real-time authentication phases.

C. Model

The Machine Learning Model forms the analytical core of the authentication system. It is built using a One-Class Support Vector Machine (SVM) from Scikit-learn, trained solely on genuine user data to define the user's normal behavioral boundaries. Unlike multi-class models that require data from several users, the One- Class SVM focuses exclusively on identifying whether the observed behavior belongs to the authorized user. During operation, the model continuously compares new input data derived from cursor and scrolling activity with the stored behavioral profile. If the input lies within the learned behavioral boundary, the session continues uninterrupted. Otherwise, it triggers alerts or security actions. The model is periodically updated to adapt to natural behavioral changes, ensuring it remains reliable and accurate over time.

D. Data Storage

The Data Storage component is responsible for maintaining both raw and processed data used by the system. All captured behavioral data, including cursor positions, speed, acceleration, and scrolling metrics, are saved in structured formats such as CSV files. These datasets are used for model training, evaluation, and behavioral trend analysis. Alongside this, the User Profile Storage maintains the trained model parameters, thresholds, and other metadata representing the genuine user's behavioral pattern. This separation of raw data and model files ensures organized management, easy retraining, and faster access during real-time authentication. It also supports future scalability if multiple user profiles are to be integrated into the system.

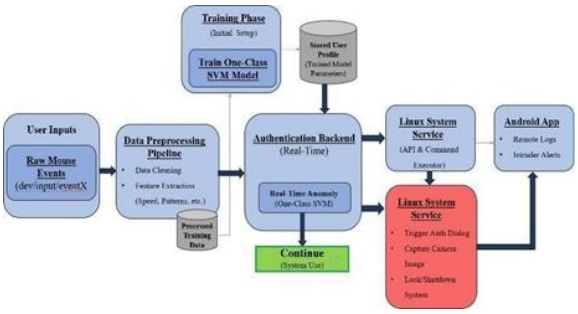
E. Supporting Scripts and Tools

The system includes several supporting scripts and configuration files that streamline execution and ensure environment consistency. The setup.sh script is responsible for automatically configuring the runtime environment by installing required dependencies and initializing system variables. The requirements.txt and config.py files specify all necessary libraries, thresholds, and authentication parameters to maintain uniform operation across different platforms. Additionally, specialized logging and reporting scripts record session events, anomalies, and authentication results in log files. These records serve as a valuable resource for analyzing system accuracy, user behavior variations, and the overall reliability of the model.

F. Security and Response Layer

The Security and Response Layer is a critical component that ensures active protection during the authentication process. When the machine learning model detects a behavioral mismatch. this module takes immediate action to safeguard the session. It can trigger real-time alerts to notify the user of suspicious activity, lock the system to prevent unauthorized access, or request re- authentication through secure methods such as a password or OTP.

In addition, it maintains detailed audit logs of all authentication events and system responses, which can later be analyzed for performance evaluation and security auditing. This layer ensures that even if the initial authentication is compromised, the system continues to provide a robust secondary defense throughout the session.



System Architecture

VII. MODULES

The proposed system for continuous behavioral authentication through cursor dynamics and scrolling behavior is divided into several interconnected modules, each responsible for a specific functionality in the overall process. These modules work together to ensure accurate data capture, feature extraction, model training, real-time authentication, and reporting, forming a complete and efficient behavioral verification framework.

The first component, the Capture Data Module, is responsible for collecting raw behavioral data from the user's interactions. It includes a Mouse Activity Logging subsystem that continuously records mouse movements, speeds, directions, clicks, drags, and scroll actions as the genuine user interacts with the system. Additionally, a Session Context Capture mechanism logs active applications and tasks to provide environmental context for each recorded behavioral pattern, ensuring that the data reflects realistic usage. The raw data is then subjected to Data Cleaning, where accidental cursor jitters, random spikes, and noise are removed to prevent distortion during analysis. Finally, Normalization and Missing Value Handling are applied to scale features uniformly and to address any incomplete or inconsistent data, ensuring the dataset's integrity and reliability.

The second component, the Feature Extraction Module, processes the pre-cleaned data to derive meaningful behavioral attributes. It focuses on two primary feature categories: Movement-Based Features and Click and Scroll Features. Movement-based features include metrics such as cursor speed, acceleration, path curvature, and movement angles, which capture unique motor characteristics of the user. The click and scroll features, on the other hand, analyze click frequency, timing intervals between clicks, scroll duration, and scroll intensity parameters that collectively model the user's interaction rhythm and scrolling habits. These extracted features form the foundation of the user's behavioral signature and serve as the input for model training. The next stage is the Model Training Module, which is crucial for building the user's behavioral model. This module employs One-Class Support Vector Machine (SVM) algorithms trained exclusively on genuine user data.

The model learns the "normal boundary" of legitimate user behavior, allowing it to detect anomalies during real- time operation. The Model Validation step evaluates the trained model's accuracy and stability, ensuring it can effectively distinguish between normal and abnormal behavioral patterns. By focusing on one-class modeling, the system eliminates the need for multi-user datasets, reducing computational complexity while improving performance in single-user authentication scenarios.

The Real-Time Authentication Module serves as the operational core of the system, continuously analyzing ongoing user behavior during active sessions. This module compares newly captured mouse and scrolls data against the trained behavioral profile to validate the user's identity in real time. The Session Validation subcomponent grants access if the behavior matches the stored profile, while deviations trigger anomaly flags that indicate potential unauthorized access. Complementing this is the Console-Based Monitoring and Response Module, which displays real-time authentication results, alerts, and status updates on the terminal. This module also handles Automated Security Actions, such as issuing alerts, locking the session, or requesting secondary authentication (e.g., password or OTP) when suspicious activity is detected.

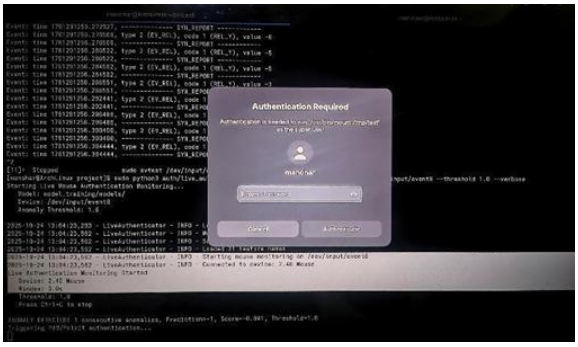
Finally, the Reporting and Audit Module provides post-authentication transparency and system accountability. It maintains Behavioral Logs, which store detailed records of captured interaction data and authentication events in structured formats like CSV, JSON, or standard log files. These logs are later utilized for Audit and Analysis, generating reports

that visualize authentication trends, detect recurring anomalies, and assess overall system performance. The insights derived from these reports assist in fine-tuning the model and improving future authentication accuracy.

Overall, these integrated modules work cohesively to ensure that the proposed system provides a robust, adaptive, and continuous authentication mechanism. Each module contributes to maintaining high accuracy, reducing false positives, and ensuring a secure yet seamless user experience without disrupting normal workflow.

VIII. RESULTS AND DISCUSSIONS

The system demonstrates a high level of effectiveness in identifying and authenticating legitimate users by analyzing their unique mouse behave or patterns. By continuously learning and recognizing subtle interaction characteristics such as movement style, speed variations, and clicking habits, it establishes a reliable behavioral profile for each user. This enables seamless and non-intrusive authentication, reducing the need for repeated manual verification while maintaining strong security.

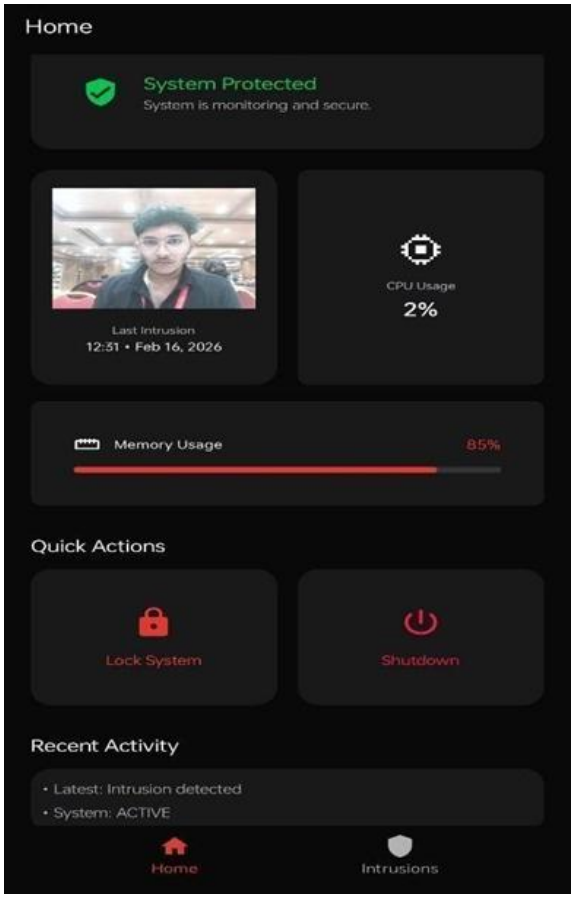


Results and Discussion

The system monitors the behavior of the user continuously. When it detects an intrusion, the system will automatically block the current user by popping the authentication alert to enter the password to get the access back to the respective user to use it. The system cannot be used and cannot shutdown until the user enters the correct password.

To further strengthen security, the system incorporates intruder detection through webcam image capture. Whenever suspicious activity is identified, the system captures an image of the user and logs it along with a precise timestamp. These images are then displayed on a web-based dashboard, allowing administrators or users to visually verify intrusion attempts. This adds an additional layer of evidence and enhances accountability by providing clear, time-stamped records of unauthorized access attempts.

The Android control application effectively integrates with the backend system to provide real-time monitoring capabilities. It successfully retrieves live system metrics such as CPU usage, memory consumption, and detected intrusion events through a dedicated API. This ensures that users can stay informed about the system's status at any time, even when they are away from their primary device. The app's responsiveness and accurate data synchronization contribute to a smooth and reliable user experience.



Results and Discussions

In addition to monitoring, the mobile application enables remote execution of critical system control commands. Actions such as system locking, user logout, and complete shutdown can be initiated directly from the app, and they execute reliably on the target system. This remote-control capability enhances both security and convenience, allowing users to take immediate action in response to suspicious activity or to manage their system efficiently from a distance.

Results and Discussions

IX. CONCLUSION

This project demonstrates an effective approach to implementing continuous user authentication based on behavioral characteristics such as cursor dynamics and scrolling behavior. Throughout the system, detailed user interactions including mouse movements, clicks, drags, and scrolls were continuously captured to form a comprehensive behavioral dataset. These captured patterns reflect the user's unique interaction tendencies, which serve as a digital behavioral signature. The raw data was carefully processed through several preprocessing steps including noise reduction, jitter smoothing, handling of missing values, and normalization to ensure consistency and accuracy. This structured preprocessing pipeline helped transform raw behavioral data into a clean, well-organized format suitable for advanced machine learning analysis. From the preprocessed data, essential behavioral features such as cursor speed, acceleration, movement angles, path curvature, click frequency, and scroll intervals were extracted. These features collectively describe the subtle motor and cognitive patterns of each user, forming the basis for behavioral identity modeling.

The extracted features were then used to train a One-Class Support Vector Machine (SVM) model, which effectively learned the genuine user's behavioral boundaries. During real-time operation, the system continuously compared live behavioral inputs against the stored user profile to verify identity, thereby enabling ongoing authentication throughout the session. The results confirm that behavioral biometrics can be used to maintain security without relying on traditional static credentials. The approach ensures that any deviation from a user's typical behavior can be quickly identified, preventing unauthorized access in real time. Moreover, the system achieves this through a non- intrusive, lightweight, and adaptive mechanism that operates seamlessly in the background, maintaining both security and usability. In conclusion, this project successfully establishes that continuous behavioral authentication using cursor dynamics and scrolling patterns can serve as a reliable, efficient, and user-friendly method for enhancing digital security. By leveraging natural user interactions, the system provides an intelligent solution capable of real-time identity verification and proactive threat detection, marking a significant step toward future adaptive cybersecurity systems.

REFERENCES

[1] Subash et al., "Adaptability of current keystroke and mouse behavioral biometrics for continuous

authentication," Computers & Security, vol. 145, 2025.

[2] "Trustworthy interaction model: Continuous authentication using time–frequency joint analysis of mouse biometrics," Behaviour & Information Technology, vol. 44, no. 2, pp. 112–126, 2025.

[3] "User authentication based on mouse dynamics using an EfficientNet model," in Proc. Int. Conf. on Artificial Intelligence and Security, 2025.

[4] S. H. Aljoubory et al., "User authentication based on mouse dynamics," Iraqi Journal for Computers and Informatics, vol. 6, no. 1, pp. 25–31, 2025.

[5] J. D. M. A. T. Vaidya, "Behavioural biometrics for continuous authentication in modern cybersecurity," International Journal of Computer Technology, vol. 18, no. 3, pp. 78–84, 2025.

[6] S. Ayeswarya et al., "Enhancing security and usability with context-aware multi-biometric continuous authentication," Scientific Reports, vol. 15, 2025.

[7] "Behavioral biometrics and continuous authentication in cybersecurity systems," ResearchGate Publication, 2025.

[8] "Behavioral biometrics for continuous authentication," International Journal of Scientific Research in Engineering and Management (IJSREM), vol. 9, no. 4, 2025.

[9] S. Abba et al., "Behavioral biometrics-powered continuous authentication for zero-trust remote work environments," Asian Journal of Research in Computer Science, vol. 18, no. 2, pp. 45–56, 2025.

[10] A. G. Uymin, "Analysis of computer mouse movements to protect remote user authentication," Journal of Information Security Studies, vol. 10, no. 1, pp. 33–40, 2025.

[11] Y. Wang et al., "Optimizing mouse dynamics for user authentication by machine learning," arXiv preprint arXiv:2504.21415, 2025.

[12] R. Dillon et al., "An agent-based modeling approach to free-text keyboard dynamics for continuous authentication," arXiv preprint arXiv:2505.05015, 2025.

[13] "Behavioral biometrics-based authentication system using machine learning," Master's thesis, Tallinn University of Technology, 2025.

[14] "Behavioral biometrics for continuous authentication using mouse and keystroke data," in Proc. Int. Conf. on Cybersecurity and Machine Learning, 2025.

[15] "AI-driven user profiling for behavioral biometric authentication," in Proc. Int. Security and Privacy Conference, 2025.

[16] "Deep learning framework for mouse dynamics-based user verification," in Proc. IEEE Int. Conf. on Biometrics and Authentication, 2025.