Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 274 (2025) 402–411

Procedia Computer Science

www.elsevier.com/locate/procedia

24th International Conference on Modelling and Applied Simulation (MAS 2025), held within the 22nd International Multidisciplinary Modeling & Simulation Multiconference (I3M 2025)

Behavioral Biometrics for Remote Exam Integrity: Continuous Authenticity Assessment via Keystroke Dynamics

Roberto Dillon [a]*, Maria De Marsico[b]

[a] *James Cook University, 149 Sims Drive, Singapore*
[b] *Sapienza University of Rome, Via Salaria 113,, Rome, Italy*

Abstract

Remote exams have become a staple in education, yet ensuring academic integrity without intrusive monitoring remains a challenge. Traditional solutions, such as webcam-based proctoring, face technical limitations and raise student anxiety. This preliminary study explores keystroke dynamics as a transparent, zero-trust approach to continuous authentication to detect impersonation during remote assessments. To this aim, it evaluates three different machine learning techniques, i.e. Random Forest, Isolation Forest, and One-Class SVM with no previously stored database of students' profiles. The lack of a users' gallery distinguishes this proposal from most literature, which deals with authentication following an explicit enrolling phase. In this study, biometric profiles are built at the beginning of the examination after the initial identification, assuming early typing patterns belong to the original account owner. Then, the corresponding profile is constantly matched against the incoming typing data to flag possible anomalies throughout the remaining part of the exam. Experiments on synthetic agent-based data (simulating both legit and cheating combination of users) yielded promising outcomes: by defining a common Risk Score (RS) metric to summarize results across all methods, all legit exams were correctly identified with no false positives (i.e. RS = 0). Random Forest and Isolation Forest detected 83% of cheating combinations (i.e. RS > 0) while OneClassSVM detected 67%. No false negatives, i.e. 100% detection of cheating instances, could be achieved only by an ensemble approach combining all the implemented techniques together and adding their respective scores. The results suggest keystroke dynamics can help identifying suspicious activity in most cases while minimizing disruptions to legitimate test-takers. Keystroke-based authentication can be a feasible and low-intrusion alternative to camera monitoring, helping institutions balance exam security with student privacy.

* Corresponding author. Tel.: +65 6709-3711
E-mail address: roberto.dillon@jcu.edu.au

1877-0509
10.1016/j.procs.2025.12.040

Keywords: keystroke dynamics; remote proctoring; agent-based modelling; random forest; isolation forest; OneClass-SVM.

1. Introduction

Security researchers have a long history of experimenting with keystroke dynamics. Beginning with the pioneering work of [1] and [2], diverse applications emerged, from fixed-text analysis for identifying users during password entry [3] to free-text methods for detecting impostors in different scenarios [4], including cases of business email compromise (BEC) [5]. In such systems, machine learning models compare one or more user profiles against new input to identify users or flag anomalies accordingly.

However, continuous biometric identification is not limited to traditional cybersecurity authentication problems, but it also holds promise for enhancing integrity in education, where remote exams have become increasingly common [6]. The COVID-19 pandemic accelerated the adoption of remote proctoring, and many universities continue to use online assessments due to their practical convenience despite concerns about cheating [7]. Current solutions to monitor examinations integrity often rely on multi-modal authentication, combining knowledge factors (e.g., passwords) with biometric verification via webcam facial recognition. Unfortunately, over 20% of students are reported to face technical barriers with camera-based systems, including hardware incompatibilities and bandwidth limitations [8]. This is also due to the fact that a single webcam is not usually deemed sufficient to ensure a fair test, and the adopted exam protocol often requires to have a further camera active to control the student's surrounding ambient. Moreover, continuous webcam monitoring is perceived as intrusive and has been linked to elevated test anxiety [9]. Last but not least, the burden of continuous monitoring also falls on the examiners.

To offer a less intrusive yet robust alternative to webcam monitoring while maintaining the zero-trust principle critical for exam integrity, keystroke dynamics verifying the ongoing input during the examination is a possible alternative approach that did not receive much attention in the past [10] until the COVID-19 pandemic forced students all over the world to learn and test remotely.

Among the latest research, the authors of [11] classified users based on the frequency of alphabetic letters in their writing and the corresponding dwell times. They implemented OneClass-SVM, KNN, and Euclidean and Manhattan metrics to detect anomalies in the users' typing rhythm, obtaining good results across all methodologies. The study tested different potential students where each profile was built beforehand by storing data from up to ten different typing sessions. Therefore, the student was already known to the system. Convolutional and Recursive Neural Networks (CNN and RNN) were also effectively employed in [12]. These studies used a character-based window, meaning that the analysis of the extracted features would start only once a sequence consisting of a certain number of characters was completed, regardless of how long it would take the user to type them. Other studies, including [13], had users typing different sessions across multiple days, to build reliable profiles that could help identify them more accurately across different conditions, settings, and even moods. This is beyond the scope of the present work, where continuous identification happens in a relatively short time and with usually little change in context, with the possible exception of mood. The latter is anyway difficult to deal with, since also past sessions could not have caught different emotional conditions. This study, on the other hand, is not concerned about identifying users in general and across multiple typing sessions but, instead, its objective is only to confirm that the specific user who truthfully identified himself to start a remote examination, is not replaced later by somebody else while the session is still in progress. For this purpose, we investigate keystroke dynamics by analyzing incoming data via a time-based window and assume no previous student biometric profile was stored prior to the examination session, making the system simpler since it does not have to identify the same user across multiple set-ups.

Three common strategies are implemented (Random Forest, Isolation Forest, and One-Class SVM) on simulated users, where each method constructs a unique biometric profile during the exam's initial typing phase, and then monitors subsequent input for anomalies. Specifically, we address the following research question:

RQ: *Can keystroke dynamics enable reliable, zero-trust continuous authentication in remote assessments by building a biometric profile at the beginning of the exam, with no previous knowledge of the candidate, and then verify it against the upcoming input?*

The paper is divided into different sections. Section 2 introduces related work and outlines key differences in the presented approach. The methodology is explained in Section 3, while results are presented in Section 4, which is followed by Sections 5 and 6 for the discussion and conclusions, respectively.

2. Related Work

In this section, we provide a short literature review about research on keystroke-based authentication, limiting its scope to approaches targeted at online examinations. Most online learning management systems require learners to log in to their own course through authentication. This can be done for personalization purposes, e.g., to create and deploy tailored learning paths, or for accounting purposes, e.g., to control the time actually spent by the student on the platform, or even to protect the teaching material from unauthorized access. This authentication can exploit either traditional methods, like passwords or tokens, or could embed biometric algorithms. Identity verification is also required during online examinations, which could be supported by dedicated platforms like, e.g. exam.net.

However, in this case specific problems arise. First, differently from other applications, a student may provide the owned credentials to another student to pass the exam. This may be mitigated by the personal check by the teacher or using biometric traits. After this initial authentication, the continuous re-assessment of the verified identity is an additional and compelling requirement. This specific problem has nurtured dedicated research. A viable and promising solution is to implement continuous implicit authentication [14]. It does not require explicit user actions, and most of all can be even carried out using the same actions that the user is doing. The fact of being transparent, unobtrusive, and not requiring to interrupt the ongoing action os particularly suited for online exams. The interested reader can find a wider survey of research proposals for online learning in [15]. A possible candidate is represented by the use of keystroke-based techniques.

The analysis of keystroke dynamics can be based on fixed-text or free-text. The first approach requires training the recognition with a predefined text, e.g. a password or passphrase, which the user must reproduce at the login time. The free-text keystroke approach does not require that the text used for enrollment and the text used for recognition/authentication are the same. The user template can be formed using timing and non-timing features. Timing features are those associated with keyboard events, namely key down and key up, that are represented by their distinct timestamps. The events can be combined to build uni-graph features like <Down-Up> that capture timing patterns of individual key presses, and bi-graph features like <Up-Down>, <Down-Down>, and <Up-Up>, which capture timing patterns corresponding to consecutive key pairs. The approach proposed in [16] exploits timing features to train a Bidirectional Long Short-Term Memory (Bi-LSTM) network. Timing features are also used in [17] where the authors purpose a Recurrent Confidence Model (R-RCM) which considers each and every action of a user in order to decide if the user is legitimate or not.

The non-timing features rather consider aspects, which are interesting as well but need sensors, usually available in mobile devices, to measure pressure, position, and finger placement. The proposal in [18] uses these features in addition to timing ones, training a MLP that is a feed forward NN. In the already mentioned approach in [12], the user keystroke data is divided according to fixed time-length windows and then converted into a keystroke vector sequence based on the time feature. A CNN-RNN model is trained for authentication. This training entails that there is an explicit enrollment phase before the continuous recognition. The approach proposed in [19] deals with assessments with Multiple Choice Questions (MCQ). It combines reinforcement of integrity policy for prevention, and keystroke-based random authentication to detect impersonation. Feature selection methods are evaluated to overcome the problem of the high-dimensional keystroke dataset. Three Machine Learning (ML) models are trained for each student using three classifiers: One-Class Support Vector Machine (SVM), Local Outlier Factor (LOF), and Isolation Forest (IF). The highest accuracy (83%) is achieved by the Isolation Forest classifier.

In the proposed paper, a critical departure from conventional keystroke dynamics systems lies in our treatment of temporal feature stability. Traditional authentication assumes medium-to-long-term permanence of behavioral traits, features must remain consistent across days, weeks, or even months to enable reliable comparison with enrolled profiles. This is impractical for continuous monitoring, where short-term stability suffices. Our approach exploits the observation that keystroke dynamics, while volatile over longer periods (due to mood, fatigue, or hardware changes), exhibit sufficient short-term consistency, especially within the confined duration of an exam session (e.g. 1 to 3

hours). Unlike enrollment-based systems, which rely on historical profiles prone to drift, we capture dynamic features at session start, when it is very infrequent that the legit user has already been substituted by an impostor. The profile is then validated against real-time input without requiring cross-session feature stability. In this way, we eliminate the need for long-term storage of templates, also reducing privacy risks in case of data breaches. By decoupling authentication from re-assessment, we address a key limitation of prior work: the assumption that typing behavior is invariant across sessions. Our method is inherently adaptive to intra-session variability while ignoring inter-session noise, a trade-off explicitly optimized for high-stakes, time-bound scenarios like exams or sensitive transactions.

3. Methodology

Following the example of previous works such as [20], we analyze keystroke dynamics in terms of five features, extracted from all keyboard characters-related events within a sliding time-based window spanning the last 5 seconds of typing and updated every second. Namely:

- Average and standard deviation of Dwell Time (DT), defined as the time in milliseconds (ms) between a key press and a key release event on the same key.
- Average and standard deviation of Flight Time (FT), defined as the time in ms between a key release event and the following key press event; considering the aggregate features allows a text-agnostic procedure.
- Percentage of errors, defined as the ratio of detected backspaces divided by the total number of key presses across the time window times 100.

For a typical user typing at a speed of 60 wpm, each time window has, on average, about five words and corresponding characters to analyze. A time window with no or only a few keypresses is discarded from analysis. These sparser time windows represent a kind of thinking time. For this experiment, a synthetic dataset was generated by using the agent-based modeling (ABM) proposed in [14] and made freely available via its online repository. The model simulates the typing of random sequences of characters with probabilities consistent with the English language and takes into consideration the keyboard type (i.e. membrane or mechanical) and the relative distance between keys. In particular, the model produces English-like character sequences using a probabilistic estimate of character frequency distribution in English text, where character c is sampled accordingly as $P(c) = fc / \sum i\ fi$ where fc represents the frequency of character c in English text and letter frequencies are normalized to sum to 1.0 Unique users can be defined by specifying different parameters such as dominant hand, error rates, fatigue factor (affecting typing speed in long typing sessions), and finger agility. Finger agility is an overall measure that refers to the speed, accuracy, and fluidity of finger movements across the keyboard during key press, and also depends on the correct use of all fingers over the keyboard.

Three different student profiles were created with the characteristics listed in Table 1. Typing agents were associated with these profiles and each agent was tasked to type in 3000 characters, roughly simulating ten minutes of actual typing for a user typing at 60 wpm where each word, on average, is made by five characters.

Table 1. Simulated users. All users were assumed to be typing on a “QWERTY” membrane

Student	WPM	Error Rate	Fatigue Factor	Finger Agility	Dominant Hand
1	78.9	0.052	$1.3\ 10^{-4}$	0.99	Left
2	41.9	0.017	$1.3\ 10^{-4}$	0.86	Right
3	58.2	0.097	$1.2\ 10^{-4}$	0.91	Right

Student 1 is supposed to be a proficient typist and is left-handed while Student 2 and 3 are right-handed and slower. They all have low fatigue factors, meaning fatigue will not affect their typing behavior during a short exam session like the one simulated in this experiment. Once the features are extracted from the simulation, effectively making a personalized exam keystroke report for each student, we combine the data to simulate exams takeover where a different student steps in while the exam is in progress.

For simplicity, we are assuming the correct student begins the exam and types for 3 minutes before the impostor steps in. The impostor is simulated by copying and pasting the respective typing data into the legit's student file.

In the following analysis, the first three minutes of typing data are used for training to define a unique profile characteristic of the legit student while the remaining data are used for verifying that the same user continues the exam as expected.

4. Results

Different combinations of students were used to compose legit and cheating sequences of typing. Each combination was tested by using Random Forest (RF), Isolation Forest (IF), and OneClass-SVM (OSVM), which are common and widely used approaches in literature for solving classification and anomaly detection problems as illustrated in Section 2. All algorithms were implemented in Python using the sklearn library and its components. For each situation, i.e. legit or cheating exam data, we take into account:

- The detected Authenticity Score (AUT), i.e. the percentage of authentic periods detected, where a "period" corresponds to the data extracted in a specific five seconds time-window.
- The longest anomaly duration (LAD) in seconds, i.e. the time a likely impostor is detected continuously.
- The number of times a detected anomaly duration lasted for more than 10 seconds (N10), i.e. 11 seconds or more.

4.1. Random Forest

Random Forest has proved itself an effective machine learning approach in keystroke dynamics scenarios [21] but it was originally designed for classification purposes so it needs to be trained with data from the different distributions it wants to classify, usually labelled as "Class 0" and "Class 1". Here, instead, we have an anomaly detection problem and no existing user profile before the exam starts. This implies that, while we train the system to recognize the original student as Class 0 by acquiring the data from the first three minutes of typing, we also have to simulate anomalies to represent different Class 1 data. The latter were created by adding controlled noise to authentic samples, for example by multiplying real samples by a factor such 0.5, 0.75, 1.5, and 2.0. The Class 1 derived dataset included at least 30% of data points originated from the incoming data.

Results for simulated regular exam sessions are presented in Table 2. Simulated cheating cases are presented in Table 3. Anomalies are detected whenever the probability of being identified as Class 0 is lower the 0.6.

Table 2. Results for AUT, LAD, N10 via Random Forest analysis in simulated legit scenarios (i.e. the legit student is the only participant).

Student	AUT	LAD	N10
1	77.4%	6	0
2	71.0%	9	0
3	73.2%	10	0

Table 3. Results for AUT, LAD, N10 via Random Forest analysis in simulated cheating scenarios (i.e. a different student steps in when the exam is in progress once the training phase is over). Rows represent the legit student (i.e. the student the system was trained upon). Columns represent the student stepping in during the exam.

Student	1	2	3
1		AUT: 71.9% LAD: 9 N10: 0	AUT: 51.5% LAD: 12 N10: 2
2	AUT: 52.3% LAD: 21 N10: 5		AUT: 44.4% LAD: 19 N10: 4
3	AUT: 69.8% LAD: 13 N10: 1	AUT: 56.8% LAD: 19 N10: 3	

Figures 1 and 2 show an example of exam progress in a legit and cheating case, respectively. Authentication probability is mapped on the y-axis and the detection threshold (dotted line) is set at 0.6 and it is plotted in red. The x-axis represents time. We can appreciate how only short anomalies are detected and overall accuracy is above 70% in the first case while several longer anomalies are detected in the second case.

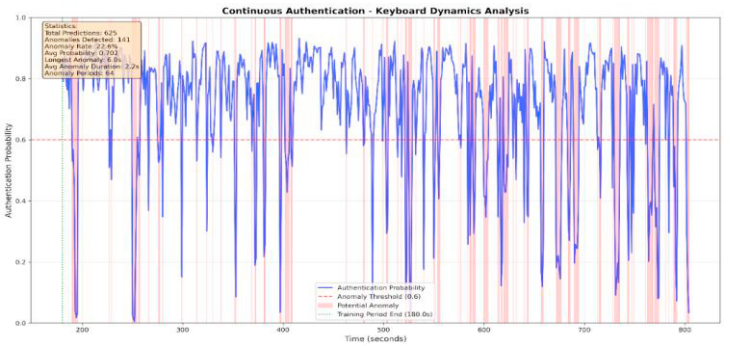

Fig.1. Continuous authentication for simulated legit exam by Student 1. Red bars identify detected anomalies. The longest detected anomaly lasts 6 seconds.

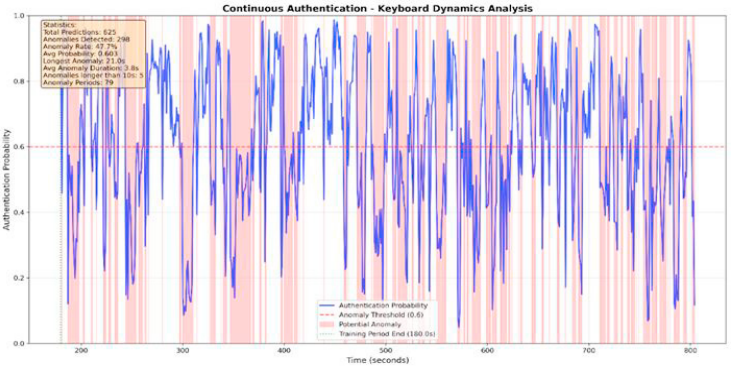

Fig. 2. Continuous authentication profile for the exam by Student 2 but with Student 1 taking his place after the exam began. Red bars identify detected anomalies. The longest detected anomaly lasts 21 seconds and there are 5 anomalies lasting more than 10 seconds.

4.2. Isolation Forest

While not as extensively adopted as Random Forest, Isolation Forest is a relatively recent unsupervised method explicitly crafted to identify anomalies within a dataset [22]. Rather than modeling normal data patterns or class distributions, Isolation Forest isolates anomalies by recursively partitioning data through random feature selection and split points, constructing binary trees where the path length required to isolate a data point serves as the anomaly score.

Anomalies, which are supposed to be relatively rare and distinct, are typically isolated with fewer splits, resulting in shorter average path lengths compared to normal instances. This approach enables Isolation Forest to efficiently detect outliers in high-dimensional and large-scale datasets, making it suitable for real-time anomaly detection scenarios such as continuous authentication in remote exams.

Results for simulated regular exam sessions are presented in Table 4. Simulated cheating cases instead are presented in Table 5.

Table 4. Results for AUT, LAD, N10 via Isolation Forest analysis in simulated legit scenarios.

Student	AUT	LAD	N10
1	80.6%	6	0
2	81.6%	9	0
3	75.8%	8	0

Table 5. Results for AUT, LAD, N10 via Random Forest analysis in simulated cheating scenarios. Rows represent the legit student while columns represent the student stepping in during the exam.

Student	1	2	3
1		AUT: 76.8% LAD: 12 N10: 2	AUT: 58.5% LAD: 17 N10: 1
2	AUT: 57.8% LAD: 14 N10: 4		AUT: 36.1% LAD: 18 N10: 12
3	AUT:80.5% LAD: 7 N10: 0	AUT: 56.0% LAD: 20 N10: 4	

Figures 3 and 4 show an example of exam progress in a legit and cheating case, respectively. The Isolation Forest score (IF) is mapped on the y axis and samples are considered legit whenever (IF) is greater than 0 (dotted red line).

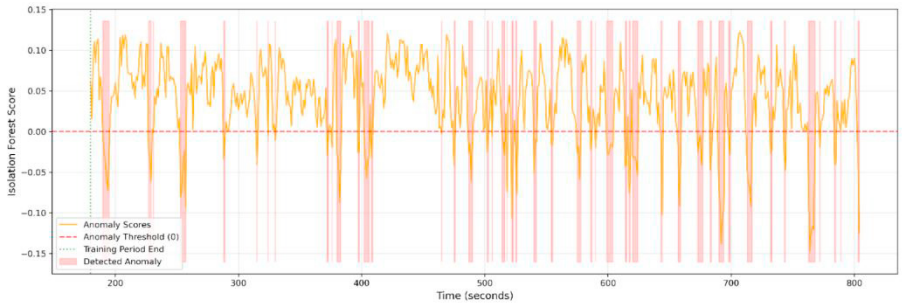

Fig. 3. Continuous authentication for simulated exam by Student 1. Red bars identify detected anomalies. The longest detected anomaly lasts 6 seconds.

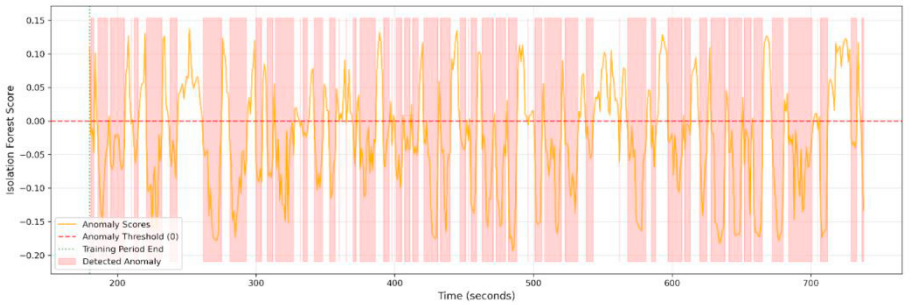

Fig. 4. Continuous authentication profile for the exam by Student 2 but with Student 3 stepping in after the initial training phase. Red bars identify detected anomalies. The longest detected anomaly lasts 18 seconds and there are 12 anomalies lasting more than 10 seconds.

4.3. OneClass-SVM

Like Isolation Forest, One-Class Support Vector Machines (One-Class SVM) also represents a specialized unsupervised learning approach suitable for cases where only legitimate user data is available during training. While reported results for keystroke analysis are not always consistent, ranging from excellent in [23] to not so good in [20], it remains one of the most natural approaches to try when experimenting with continuous authentication systems. Results for simulated regular exam sessions are presented in Table 6. Simulated cheating cases instead are presented in Table 7.

Table 6. Results for AUT, LAD, N10 via OneClass-SVM analysis in simulated legit scenarios.

Student	AUT	LAD	N10
1	77.9%	9	0
2	83.2%	9	0
3	80.1%	8	0

Table 7. Results for AUT, LAD, N10 via OneClass-SVM analysis in simulated cheating scenarios. Rows represent the legit student while columns represent the student stepping in during the exam.

Student	1	2	3
1		AUT: 76.0% LAD: 12 N10: 2	AUT: 70.3% LAD: 8 N10: 0
2	AUT: 53.1% LAD: 14 N10: 7		AUT: 41.5% LAD: 20 N10: 15
3	AUT: 81.6% LAD: 9 N10: 0	AUT: 48% LAD: 22 N10: 7	

Figures 5 and 6 show an example of exam progress in a legit and cheating case, respectively. The SVM score is on the y-axis and samples are considered legit when the score is greater than 0.

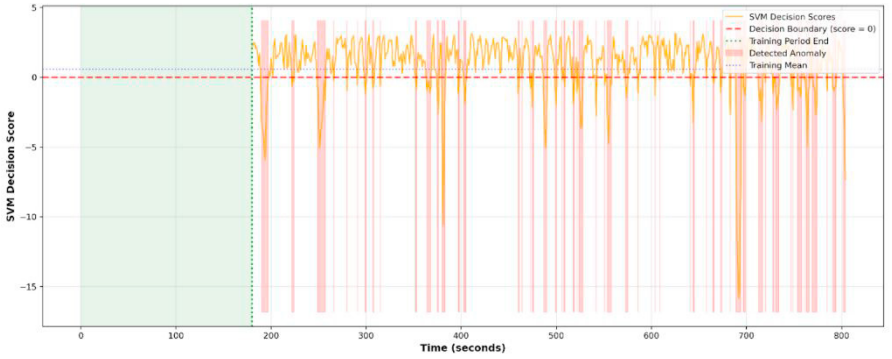

Fig. 5. Continuous authentication for simulated exam by Student 1. Decision boundary is set to 0, meaning that a decision score greater than 0 identifies a legit user. Red bars identify detected anomalies. The longest detected anomaly lasts 9 seconds.

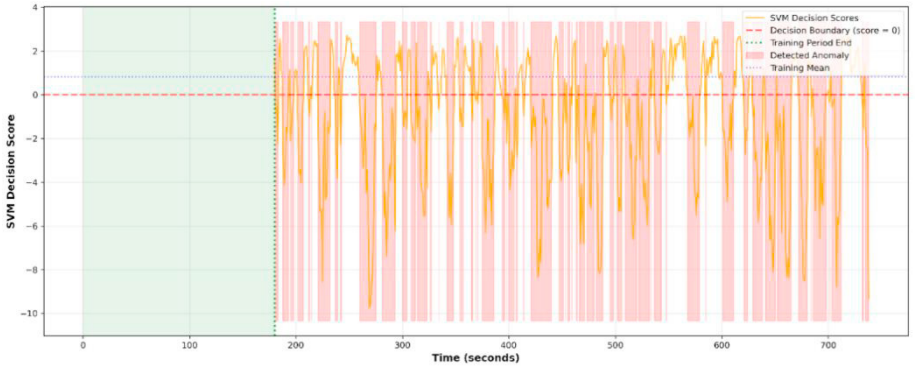

Fig. 6. Continuous authentication profile for the exam by Student 2 but with Student 3 stepping in after the initial training phase. Red bars identify detected anomalies. The longest detected anomaly lasts 20 seconds and there are 15 anomalies lasting more than 10 seconds.

5. Discussion

By looking at the results, an Authenticity threshold of 70% as well as an Anomaly period of 10 seconds appear as realistic discriminating factors between legit and possibly cheating scenarios. As empirical guidelines, then, we can assume an exam is legit as long as the overall Authenticity Score is above 70% and only short anomaly periods lasting 10 seconds or less are identified. Nonetheless, it has to be highlighted how a high value of the Authenticity Score across the overall exam session is not necessarily a proof of an honest attempt since an impostor may step in

only to answer a few specific questions in a quiz-like examination scenario, maintaining the overall accuracy rate relatively high. Hence, the detection of long, uninterrupted anomaly periods may be more significant in detecting suspicious cases in a real setting. To better compare the performance of each methodology, we can define a Risk Score (RS) by taking into account these thresholds:

$$RS = AS + DS + CS = \max\left(0, \left\lceil \frac{70 - ACC}{5} \right\rceil\right) + \max\left(0, \left\lceil \frac{LAD - 10}{2} \right\rceil\right) + n_{10} \quad (1)$$

Where:

- AS represents the "Authenticity Score", adding 1 for every 5% points less than the 70% threshold, rounded up. For example, a reported AUT of 64% would add 2 to RS.
- DS represents the "Duration Score", adding 1 for every two second above the original 10 seconds threshold of the longest anomaly period detected, rounded up. For example, a LAD of 11 seconds would add 1 to the RS while a LAD lasting 15 seconds would add 3.
- CS represents a "Counted Anomalies Score" and is simply represented by n_{10}, i.e. the number of detected anomalies lasting more than 10 seconds.

A fully legit exam should have an RS equal to 0 while an exam flagged as suspicious of containing possible cheating should have a positive score.
We can now classify and compare the previous results by scoring them accordingly (Table 8).

Table 8. Risk Scores for all the previously discussed scenarios across the three analysis methodologies (RS, IF, OSVM). Legit exams are in bold font. There are no false positives. False negatives are identified in bold and Italic font.

Student	1	2	3
1	**RS: 0** **IF: 0** **OSVM: 0**	***RS: 0*** IF: 3 OSVM: 3	RS: 7 IF: 8 ***OSVM: 0***
2	RS: 15 IF: 9 OSVM: 13	**RS: 0** **IF: 0** **OSVM: 0**	RS: 15 IF: 23 OSVM: 26
3	RS: 4 ***IF: 0*** ***OSVM: 0***	RS: 11 IF: 12 OSVM: 23	**RS: 0** **IF: 0** **OSVM: 0**

From Table 8 we see that there were no false positives across all RS, IF and OSVM analyses as all legit exams had indeed an RS equal to 0 since all parameters remained within the predefined boundaries. Each strategy performed well overall, though none was single-handedly able to detect all possible cheating combinations, which we can flag only if we adopt an ensemble approach and combine results from each technique together. In fact, Random Forest considered the case where Student 2 stepped in to support Student 1 as legit, without reporting any anomalies, while Isolation Forest did the same for Student 1 taking over Student 3. In the end both identified 5 cases correctly out of 6 (i.e. 83%). Interestingly, OneClass-SVM was able to identify some cheating cases more clearly than any other methodology, scoring two exams higher than 20, but failed to identify 2 cheating instances out of six, i.e. Student 3 taking over Student 1 and Student 1 taking over Student 3. The latter scenario was identified only by Random Forest.

6. Conclusions

The findings of this exploratory study demonstrate the feasibility of keystroke-based authentication as a low-intrusion alternative to camera monitoring, aligning with zero-trust security principles. Results from such authentication systems can support informed decisions about the integrity of online exams and effectively help teachers flag suspicious cases. By scoring exam profiles with a Risk Score metric, taking into consideration the authentication score and the presence of long anomalies, Random Forest, Isolation Forest and OneClass-SVM were all able to confirm legit exam simulations 100% of the time. When identifying cheating cases, instead, no specific method stood out by itself: RS and IF provided similar performance (83%) while OSVM missed an additional

cheating case scoring only 66.6% even though it was able to identify other cheating scenarios more clearly than the other algorithms. Nonetheless, no exam cheating scenario managed to pass undetected the scrutiny of every single analysis combined, meaning that an ensemble approach could be a viable option for identifying cheating cases that may manage to trigger a false negative by one specific algorithm. Despite the encouraging results of this pilot study, its reliance on a small, synthetic dataset obtained via agent-based modelling (ABM) simulating English-speaking students, is an obvious limitation that highlights the need for further validation and fine-tuning on real-world typing and exam data. Future work should expand to larger, diverse datasets with real users and exam durations, also investigating adaptive thresholds for anomaly detection and target accuracy, possibly depending on exam type (e.g. short answers, essay, etc.) and duration.

References

[1] Gaines, R.S., Lisowski, W., Press, S.J., and Shapiro, N.(1980) “Authenticity By Keystroke Timing: Some Preliminary Results”, Rand Report R-256-Nsf, Rand Corporation.

[2] Bleha, S. (1988) “Recognition Systems Based on Keystroke Dynamics”, Ph.D. Dissertation, University of Missouri-Colombia.

[3] Sogukpinar, I., and Yalcin, N. (2004) “User Identification at Logon via Keystroke Dynamics”, Journal Of Electrical & Electronics Engineering 4(1), 995-1005.

[4] Shadman, R., Wahab, A.A., Manno, M., Lukaszewski, M.S., Hou, D., and Hussain, F. (2023) “Keystroke dynamics: concepts, techniques, and applications”. arXiv.

[5] Dillon, R. (2024) “Who’s typing? An experiment on keyboard dynamics for BEC detection”. 15th Int Workshop Appl Modelling and Simulation (WAMS); 29–34.

[6] Portugal, D., Faria, J.N., Belk, M., et al (2023) “Continuous user identification in distance learning: a recent technology perspective”. Smart Learn. Environ, 10(38).

[7] Grove, J. (2025) “Most Universities Still Use Online Exams Despite Cheating Fears, Times Higher Education”, Times Higher Education, Available Online: https://www.timeshighereducation.com/news/most-universities-still-use-online-exams-despite-cheating-fears

[8] Turani, A.A., Alkhateeb, J. H., and Alsewari, A. A. (2020) “Students Online Exam Proctoring: A Case Study Using 360 Degree Security Cameras”. Emerging Technology in Computing, Communication and Electronics (ETCCE), 1-5.

[9] Woldeab, D., and Brothen, T.(2021) “Video Surveillance of Online Exam Proctoring: Exam Anxiety and Student Performance”, IJEDE, 36(1).

[10] Pisani, P.H. and Lorena, A.C. (2013) “A systematic review on keystroke dynamics”, *J. Braz. Comput. Soc*, 19(4), 573–587.

[11] Kochegurova, E.A., and Zateev, R.P. (2022) “Hidden Monitoring Based on Keystroke Dynamics in Online Examination System”. *Program Comput Soft,* 48, 385–398.

[12] Lu, X.F., Zhang, S.F., Pan Hui, P., and Lio, P. (2020) “Continuous authentication by free-text keystroke based on CNN and RNN”, *Computers & Security*, 96.

[13] Bours, P., and Mondal, S. (2015) “Continuous Authenticity with Keystroke Dynamics”. In: Y. Zhong and Y. Deng (Eds.): *Recent Advances in User Authenticity Using Keystroke Dynamics Biometrics.*, Ch.3, 41-58.

[14] Baig, A.F., and Eskeland, S. (2021) “Security, Privacy, and Usability in Continuous Authentication: A Survey”, *Sensors*, 21(17):5967. doi: 10.3390/s21175967.

[15] Ryu, R., Yeom, S., Herbert, D., and Dermoudy, J. (2023) “A comprehensive survey of context-aware continuous implicit authentication in online learning environments”. IEEE Access, 11, 24561-24573.

[16] Yang, L., Li, C., You, R., Tu, B., and Li, L. (2021) “TKCA: a timely keystroke-based continuous user authentication with short keystroke sequence in uncontrolled settings”. *Cybersecurity*. doi:10.1186/s42400-021-00075-9.

[17] Kiyani, A.T., Lasebae, A., Ali, K., Rehman, M.U., and Haq, B. (2020) “Continuous user authentication featuring keystroke dynamics based on robust recurrent confidence model and ensemble learning approach”. IEEE Access. 8, doi:10.1109/ACCESS.2020.3019467

[18] Salem, A., and Obaidat, M. S. (2019) “A novel security scheme for behavioral authentication systems based on keystroke dynamics. *Security and Privacy*, *2*(2).

[19] Garg, M., and Goel, A. (2025) “A comprehensive approach for mitigating impersonation in online assessment: integrity policy and random authentication”. *International Journal of Information Security*, 24(1), 1

[20] Dillon, R., and Arushi (2025) “An Agent-Based Modeling Approach to Free-Text Keyboard Dynamics for Continuous Authenticity”. arXiv preprint, arXiv:2505.05015

[21] Zeid, S., El Kamar, R.A., and Hassan, I.S. (2022) “Fixed-text vs. free-text keystroke dynamics for user authentication”, Eng Res J (Shoubra), 51(1), 95–104.

[22] Liu, F.T. , Ting, K.M., and Zhou, Z.H. (2012) “Isolation-based anomaly detection”, *ACM Transactions on Knowledge Discovery from Data*, 6, 1–39.

[23] Çeker, H., and Upadhyaya, S. (2016) “User authentication with keystroke dynamics in long-text data”, IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS).