Software Impacts 14 (2022) 100443

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

CyberSignature: A user authentication tool based on behavioural biometrics

Nonso Nnamoko *, Ioannis Korkontzelos, Joseph Barrowclough, Mark Liptrott

Department of Computer Science, Edge Hill University, St Helens Road, Ormskirk, L39 4QP, United Kingdom

ARTICLE INFO

Keywords:
Behavioural biometrics
Payment authentication
Digital identity
Cyber-security
Identity fraud detection
Machine learning

ABSTRACT

Behavioural biometrics, such as the way people type on computer keyboard and/or move the cursor are almost impossible to steal. This paper presents CyberSignature[1], a tool that uses behavioural biometrics to create unique digital identities that can be used during online card transactions to distinguish legitimate users from fraudsters. The tool is implemented in Python, with a machine learning algorithm at its core. It receives user input data entries from a graphical user interface, similar to an online payment form, and transforms them into unique digital identities. The tool is freely available on Github and is entitled 'CyberSignature'.

Code metadata

Current code version	v1.0
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2022-252
Permanent link to Reproducible Capsule	
Legal Code License	MIT License
Code versioning system used	GIT
Software code languages, tools, and services used	Python
Compilation requirements, operating environments & dependencies	Python ≥ v3.8, Kivy ≥ v2.0, numpy ≥ v1.21, pandas ≥ v1.4, scikit-learn ≥ v1.1
If available Link to developer documentation/manual	README available at: https://github.com/CyberSignature-EHU/CyberSignature
Support email for questions	nnamokon@edgehill.ac.uk

1. Introduction

Behavioural biometrics (BB) is an emerging research field focussing on methods for capturing uniquely identifying and measurable patterns in human activities, such as keystroke and mouse use characteristics, commonly known as keystroke, mouse and touchscreen (KMT) dynamics. Unlike physiological biometrics, e.g., fingerprints or iris patterns, that require active user engagement with expensive sensors or biometric scanners, KMT dynamics are gathered passively with existing hardware, such as a keyboard and mouse. This paper presents CyberSignature, a software that uses KMT dynamics to provide secure payment authentication, without compromising the user experience. CyberSignature captures user KMT dynamics, whilst users enter card details on a payment form. Then, measurable patterns generated from KMT dynamics are processed using a machine learning algorithm to generate a uniquely identifying model for the user. The model can be tested subsequently with KMT dynamics input from the same and/or other user(s) and authentication is either approved or denied.

CyberSignature is specifically relevant to online payment services in Europe, which must conform to the strong customer authentication (SCA) requirement under the second payment services directive (PSD2) [1]. The typical online payment scenario requires a user to submit their card details via a payment form and SCA requires customers to identify themselves with extra information, beyond just what they know e.g., password. When making a payment under SCA, potential customers will need to use data from two out of the following three categories to identify themselves:

- inherent data, e.g., BB, fingerprints, iris, voice, etc.
- knowledge, e.g., password, PIN etc

* Corresponding author.
E-mail addresses: nnamokon@edgehill.ac.uk (N. Nnamoko), Yannis.Korkontzelos@edgehill.ac.uk (I. Korkontzelos), barrowcj@edgehill.ac.uk (J. Barrowclough), liptrotm@edgehill.ac.uk (M. Liptrott).

[1] LinkedIn and Twitter: @cybersignature

https://doi.org/10.1016/j.simpa.2022.100443
Received 31 October 2022; Accepted 4 November 2022

- possession, e.g., one-time-password sent to phone

The traditional combination uses data based on possession and knowledge, which may cause delays and some customers may abandon their shopping carts. However, new authentication methods now incorporate inherent factors that are predominantly collected via mobile devices such as fingerprints, iris and voice. For example, after entering their card details, customers are directed to an authentication page on their card issuer's website, where they identify themselves with fingerprint and either a password associated with the card or a code sent to their phone. To the best of our knowledge, KMT dynamics has not been implemented as an inherent factor in online card payment authentication. In this paper, we demonstrate how KMT dynamics can be captured from an online payment form and used for authentication. Specifically, we present CyberSignature as a stand-alone application that wraps the core components of the typical online payment authentication process, where a user submits their card details via a payment form for authentication.

2. Software architecture

Fig. 1 illustrates the system architecture and the data flow in CyberSignature. The system consists of three main components:

(a) the **graphic user interface**, for capturing user KMT dynamics during data entry
(b) the **data pre-processing** unit, for generating unique user patterns (features) from KMT dynamics
(c) the **machine learning** unit, which learns from the features, how to make predictions on new input data

3. Software algorithm & methodology

The software algorithm captures and stores the unique characteristics of a user, so that when authentication is needed, new data is captured and compared to the stored record. The identity is confirmed, if the data matches and rejected otherwise. As shown in the CyberSignature architecture (see Fig. 1), a machine learning model is trained and stored for making such predictions on new data. The model requires ten data samples from the legitimate card owner (true data) and ten from fraudsters (false data). Thus, we *ensured* that the CyberSignature software presented in this paper is pre-loaded with ten **pre-processed** false data captured from multiple users entering a fictitious card detail displayed on the **graphical user interface**. Then, the current user is *required* to enter the same fictitious card detail ten times on the **graphical user interface**, which forms the true data that is combined with the pre-loaded false data to train and store a model. Subsequent entries of the same fictitious card details (irrespective of the user) are regarded as test data and used by the model to predict whether it is the user who entered the true data or not. This illustration of the software from a user's view point is shown in Algorithm 1 and a detailed demonstration is presented in Section 4.

CyberSignature is implemented in Python and designed to run on a laptop or personal computer with Microsoft Windows operating system installed. Detailed development methods adopted for individual components of the software are provided in Sections 3.1–3.3.

3.1. User interface

The user interface is designed to mirror a standard online payment form. We used the `Kivy`[2] Python library to capture user events related to KMT dynamics from the form's entry fields. The data is stored in JavaScript Object Notation (JSON) format, which provides convenience in working with Python due to its native JSON support. In fact, JSON files containing 1760 instances of raw KMT dynamics has been made publicly available [2,3]. These were captured from 88 consenting individuals (male or female, aged ≥ 18), whilst they entered fictitious card details onto the CyberSignature user interface.

[2] `Kivy` is available at: www.github.com/kivy/kivy

Algorithm 1 Application algorithm for CyberSignature

Ensure: trainData ← falseData ▷ falseData contains 10 pre-existing entries from multiple users
Require: trainData ← trueData ▷ trueData must be from a single user
  $newData$ = input('enter card details')
  $trueData \leftarrow newData$ from 1 to 10
  $testData \leftarrow newData$ from 11 to ∞
  if $model \leftarrow$ False **then**
    $trainData \leftarrow trueData$
    do feature engineer $trainData$
    do train $model$
  else
    if $model \leftarrow$ True **then**
      do feature engineer $testData$
      do test $model$
    end if
  end if

3.2. Data pre-processing

A wide variety of KMT dynamics is collected per user session from keyboard (e.g., 'pressed' and 'released') and mouse (e.g., 'movement', 'left press', 'left release', 'right press', 'right release', 'scroll up', and 'scroll down'). It is also possible to capture KMT dynamics from other user interface types, such as touch screens, but this is beyond the scope of this paper. The captured KMT dynamics were transformed into a reduced set of features by calculating various functions, such as minimum, maximum and mean, per test session. From these, we determined the optimal subset of features, through a technique commonly known as feature selection [4]. The selected features were used subsequently to train a machine learning model for prediction. The selected features are the mean time of keyboard dwell and the mean flight time (Fig. 2(a)); as well as the mean mouse trajectory per KMT dynamics test session (Fig. 2(b)). For example, the mean mouse trajectory in Fig. 2(b) is the total distance ($d_1 + d_2 + d_3$) over the number of pauses beyond 500 ms from the beginning to and including the end (i.e., 3 times, in this case). The selected features are stored in a `Pandas`[3] frame ready for training and/or testing machine learning models.

3.3. Machine learning

In this CyberSignature version, for the classification task, we used the `scikit-learn`[4] implementation of a simple Gaussian Naïve Bayes classifier [5] in its default settings. We considered only the three selected features, presented in Fig. 2, collected over ten user data entry sessions, to demonstrate how CyberSignature builds a classifier that can be used for user authentication.

4. Software demonstration

The user interface displays the details of a fictitious card with instructions to enter these details on the payment form (see Fig. 3). As described in Algorithm 1, CyberSignature is preloaded with 10 sets of *false* user KMT dynamic instances collected from multiple individuals entering the 'fictitious' card details displayed on the user interface. Upon initial instantiation, the software requires the *true* user to enter the same card details 10 times (Fig. 3(a)). A countdown is displayed on the user interface during *true* data entry (e.g., 'Training data entered: 1/10'). The Reset button can be used to clear the entry fields at any time, e.g., if the user makes a mistake. Both Delete Model and Save Model buttons remain inactive until 10 *true* user training data

[3] `Pandas` is available at: https://pandas.pydata.org/
[4] `scikit-learn` is available at: https://scikit-learn.org/stable/

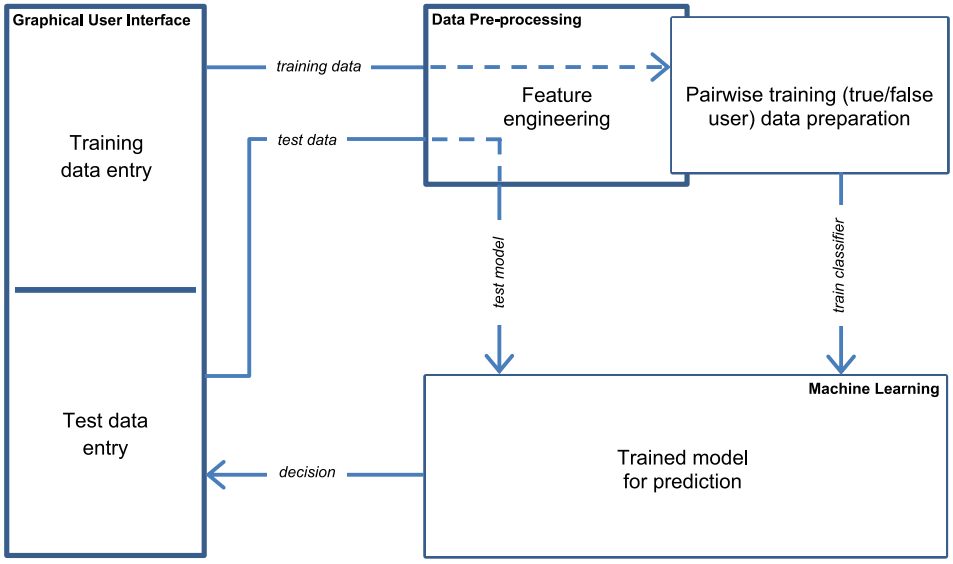
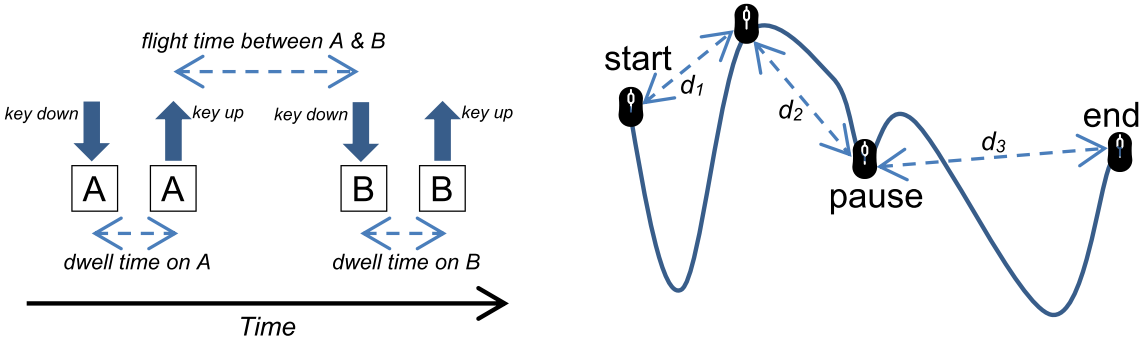


Fig. 1. CyberSignature system architecture and data flow.



(a) Keystroke features (b) Mouse trajectory features

Fig. 2. Features generated from KMT dynamics.

entries have been completed. Then, KMT dynamics features are generated from these entries and the pre-loaded *false* user data. The features are in turn used to train a classifier model, which can be saved in Sparse Allele Vectors (SAV) file format via the Save Model button.

With a saved model in place, subsequent entries of the same card details displayed on the user interface are considered *test* data instances (Fig. 3(b)) that are used to test the model's authentication accuracy in distinguishing between fraudsters (Fig. 3(c)) and legitimate users/card owners (Fig. 3(d)). The saved model can be deleted at any time using the Delete Model button, to reset the software to its original state.

An explanatory video demonstrating the training and testing procedure is available on Youtube[5]. The software source code and documentation to replicate the procedure is also available on Github[6].

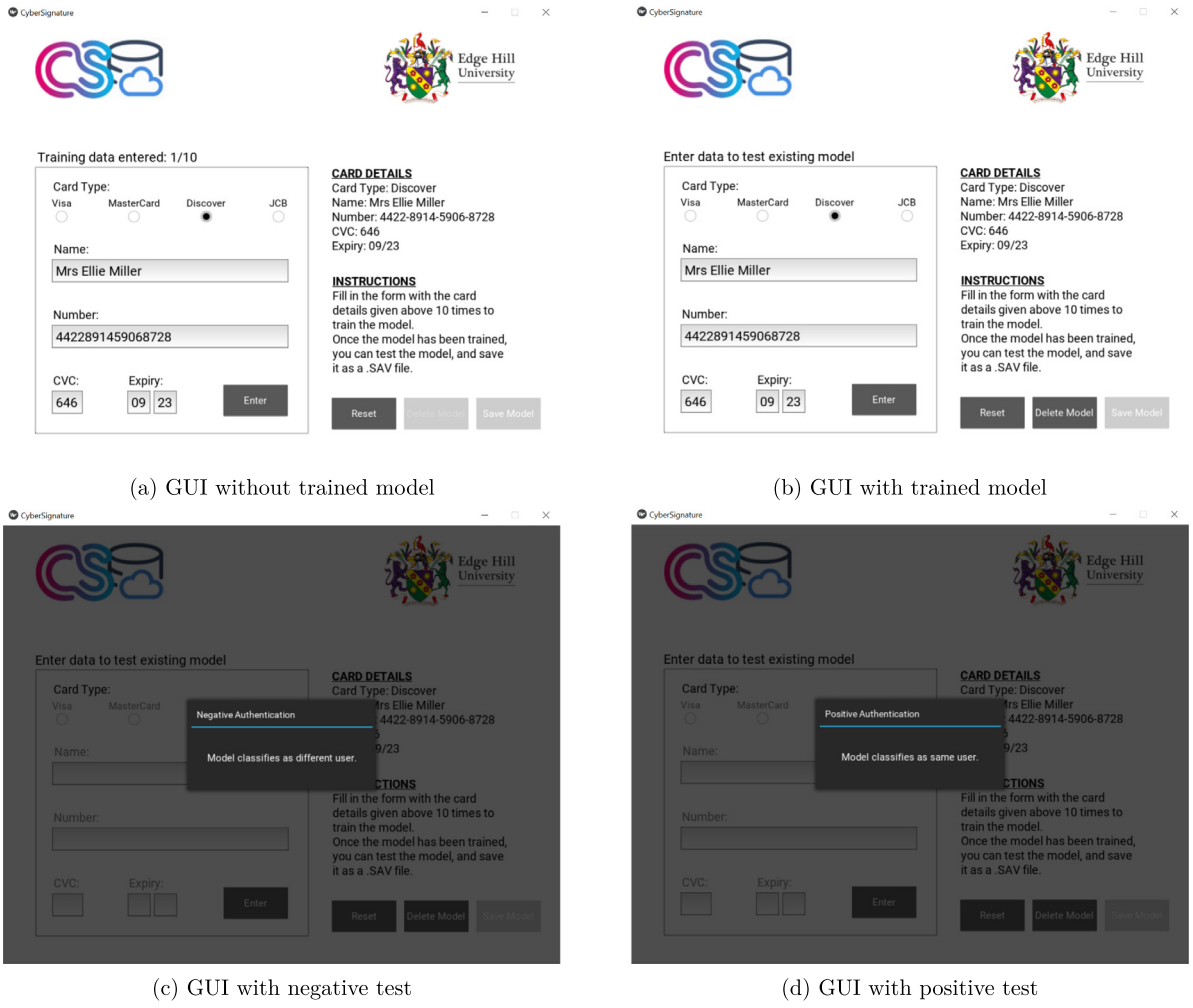
5. Software impacts

Research: While behavioural biometrics research involving KMT dynamics has had success and impact for authentication, existing implementations have focused on standalone applications, where the user needs to download and install them on their system (computer or mobile device) before running them. We argue that KMT authentication benefits are extendable to web-based application areas, hence the focus on online payment forms. The complete system architecture of CyberSignature is intended to help online payment services meet the PSD2 requirements [1] with frictionless second factor authentication. CyberSignature can be easily adapted to the payment system environment and the full implementation architecture will be published separately. It is important to note that CyberSignature is the only tool that attempts to implement KMT dynamics as an inherent factor for online payment authentication. In this paper, we presented the software as a stand-alone application for demonstration purposes. However, it sets the scene for how KMT dynamics can be utilised in a real-world payment gateway to authenticate a card user.

Industry: CyberSignature has been successfully tested with real user data with authentication accuracy up to 100% [3]. It is important to note that CyberSignature was 1 of 24 academic research projects selected by the Cyber Security Academic Startup Accelerator Programme (CyberASAP) for commercial development into the United Kingdom's (UK) £8.9billion cyber-security sector [6]. CyberASAP was cited as a successful Case Study in the UK Government's Research and Development Roadmap [7] and a video demonstrator of CyberSignature is featured on their official website [8].

[5] *CyberSignature* demonstrator video is available at: https://youtu.be/CaVNFbvKQgo

[6] *CyberSignature* is available at: https://github.com/CyberSignature-EHU



(a) GUI without trained model

(b) GUI with trained model

(c) GUI with negative test

(d) GUI with positive test

Fig. 3. Graphical User interface of CyberSignature.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research has been carried out as part of the CyberSignature Project, which received two rounds of funding from Innovate UK under the CyberASAP programme with Project Reference No. 10017354 and 10002115. Special thanks to KTN who facilitated the project delivery and to the Computer Science Department at Edge Hill University, for providing time and resources to complete this research. The authors would also like to acknowledge participants who contributed KMT dynamics dataset for the software development and validation.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.simpa.2022.100443.

References

[1] The European Parliament and the Council of the European Union, Directive (EU) 2015/2366 of the European parliament and of the council, Off. J. Eur. Union (2015) URL https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32015L2366&from=EN.

[2] N. Nnamoko, J. Barrowclough, M. Liptrott, I. Korkontzelos, Behaviour biometrics dataset, Mendeley Data v1 (2022) URL http://dx.doi.org/10.17632/fnf8b85kr6.1.

[3] N. Nnamoko, J. Barrowclough, M. Liptrott, I. Korkontzelos, A behaviour biometrics dataset for user identification and authentication, Data in Brief (2022) 108728, URL http://dx.doi.org/10.1016/j.dib.2022.108728.

[4] N. Nnamoko, F. Arshad, D. England, J. Vora, J. Norman, Evaluation of filter and wrapper methods for feature selection in supervised machine learning, in: PGNET, 2014, pp. 63–67.

[5] G.H. John, P. Langley, Estimating continuous distributions in Bayesian classifiers, in: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, UAI '95, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN: 1-55860-385-9, 1995, pp. 338–345, URL http://dl.acm.org/citation.cfm?id=2074158.2074196.

[6] KTN, Revealed: the 24 innovations selected for commercial development into the UK's £8.9billion cybersecurity sector, 2021, URL https://ktn-uk.org/news/revealed-the-24-innovations-selected-for-commercial-development-into-the-uks-8-9billion-cybersecurity-sector/.

[7] A. Sharma, UK Research and Development Roadmap, in: Policy Paper, Department for Business, Energy & Industrial Strategy, 2020.

[8] KTN, CyberSignature, in: Projects, 2022, URL https://ktn-uk.org/projects/cybersignature/.