

UL-KEYSTROKE: A WEB-BASED KEYSTROKE DYNAMICS DATASET

ARON LO LI

20160795@aloe.ulima.edu.pe

<https://orcid.org/0009-0006-2616-3950>

Universidad de Lima

JUAN GUTIÉRREZ-CÁRDENAS

jmgutier@ulima.edu.pe

<https://orcid.org/0000-0003-2566-4690>

Universidad de Lima

VICTOR H. AYMA

vh.aymaq@up.edu.pe

<https://orcid.org/0000-0002-0284-2610>

Universidad del Pacífico

Received: March 13th, 2024 / Accepted: May 23rd, 2024

doi: <https://doi.org/10.26439/interfases2024.n19.7009>

ABSTRACT. Keystroke dynamics-based authentication systems identify individuals by analyzing their keystroke patterns when interacting with input devices such as a computer keyboard. Within the fields of Statistics and Machine Learning, several research studies have applied different techniques for recognizing keystroke patterns. This work proposes the creation of a dataset and a methodology that would allow users to capture typing patterns from students at a university in Lima, Peru, using a cloud environment and their personal devices. The cloud architecture used for the implementation and deployment of the web tool will be explained in detail. The result of this work is a dataset containing participant information, records of their keystroke patterns, and additional metadata from their web browsers, which could be used to enrich further studies. Moreover, in addition to the captured raw data, some keystroke dynamics features were generated and made available along with the dataset to facilitate the development of classification models. The dataset and methodology presented in this article can be used by other researchers to enhance existing keystroke dynamics recognition systems.

KEYWORDS: keystroke dynamics / machine learning / dataset

UL-KEYSTROKE: UN CONJUNTO DE DATOS DE DINÁMICA DE TECLADO BASADO EN LA WEB

RESUMEN. Los sistemas de autenticación basados en la dinámica de teclado identifican a las personas analizando sus patrones de tecleo cuando interactúan con dispositivos de entrada, como un teclado de computadora. En los campos de Estadística y Aprendizaje Automático, existen varios estudios de investigación que han aplicado diferentes técnicas para el reconocimiento de patrones de tecleo. En este trabajo, se propuso la creación de un conjunto de datos, así como una metodología que permitiría a los usuarios capturar patrones de tecleo de estudiantes pertenecientes a una universidad en Lima, Perú, a través de un entorno en la nube y desde sus propios dispositivos. La arquitectura en la nube utilizada para la implementación y despliegue de la herramienta web será explicada en detalle. El resultado de este trabajo es un conjunto de datos con información de los participantes, registros de sus patrones de tecleo y metadatos adicionales de los navegadores web de los participantes que podrían usarse para enriquecer futuros estudios. Además, junto con los datos sin procesar capturados, se generaron algunas características de la dinámica de tecleo y se pusieron a disposición junto con el conjunto de datos para facilitar la generación de modelos de clasificación. El conjunto de datos y la metodología presentados en este artículo pueden ser utilizados por otros investigadores para mejorar los sistemas de reconocimiento de dinámica de teclado actuales.

PALABRAS CLAVE: dinámica de teclado / aprendizaje automático / conjunto de datos

1. INTRODUCTION

This article aims to comprehensively detail all aspects related to a dataset, encompassing the collection methodology, the analytical procedures conducted, the characteristics of the data, the value added by the creation of this dataset, and its potential limitations. These points will be discussed in the following sections.

2. SPECIFICATIONS TABLE

Subject	Applied Machine Learning
Specific subject area	Keystroke Dynamics Authentication
Type of data	<p>Raw data:</p> <ul style="list-style-type: none"> Records of keystroke patterns from college students (participants) and metadata of the students' web browsers used to register the keystroke patterns. List of participants along with their personal data, as well as the chosen username and password. <p>Processed data:</p> <ul style="list-style-type: none"> Time vectors generated from raw data.
Data collection	<p>A web logging application, written in JavaScript, was implemented and deployed in a cloud environment. The participants were invited to cooperate virtually in the keystroke recording sessions using their own computers, where the application captured the timestamps of each pressed and released key when typing their login credentials.</p> <p>The participants in each session were asked to perform three authentication cycles: one cycle involving the use of their own credentials and the remaining two using the credentials of randomly selected participants.</p> <p>At the end of each session, the captured records were automatically sent to a server for storage in a non-relational database.</p>
Data source location	Cloud-based collection (Heroku, MongoDB Atlas)
Data accessibility	<p>Repository name: UL-Keystroke Dynamics Dataset</p> <p>Data identification number (doi): 10.17632/9cg3c8jkh8.1</p> <p>Direct URL to data: https://data.mendeley.com/preview/9cg3c8jkh8?a=f4e49f3e-b689-4b6c-95a3-3bf5207d1935</p>
Related research article	None

3. VALUE OF THE DATA

- The scientific community has very few web-based keystroke dynamics datasets that are publicly accessible, have undergone a rigorous collection process, and have been created in uncontrolled environments.
- The published dataset contains both records of users' typing sessions as well as some metadata captured from their web browser environment. This dataset enables

researchers to leverage the gathered data to produce keystroke-related features, enhancing the effectiveness of the authentication models on which they are working.

- The generated dataset has unique characteristics compared to other public datasets. For instance, participants entered two values in the web application: a username and a password. Both values and subsequent keystrokes were captured. Unlike other datasets where participants input imposed fixed-length passwords, here the participants were given the choice to register their own password. Thus, two user groups were defined at the time of registration: some participants could choose their password but with a fixed length, while others had no length restriction.
- This keystroke dynamics dataset will allow researchers to strengthen ongoing core authentication systems by adding an additional layer of security through the application of Deep Learning and Machine Learning models or similar, which can be applied in critical systems such as banking, medical care, military, etc.

4. BACKGROUND

In the field of automatic authentication, several works have focused extensively on generating keystroke dynamics models. However, it is also essential to have a dataset with relevant information about keystroke patterns to generate a high-quality authentication model. In the literature, there are only a few published datasets available for researchers (Giot et al., 2009; Killourhy et al., 2009). This work aims to provide the public with access to a keystroke dynamics dataset generated from a cloud-based web approach, as well as the methodology used to construct it. This will enable future researchers to understand how to generate their own keystroke dynamics datasets and serve as a reference for conducting their collection.

5. DATA DESCRIPTION

The dataset comprises a collection of keystroke samples from various participants who voluntarily took part in the data collection process (experiment). These samples were captured using a web application, implemented and deployed in a cloud environment. This setup allowed the participants to conduct the tests on their computers from any location, ensuring that the experiment took place in real-world environments.

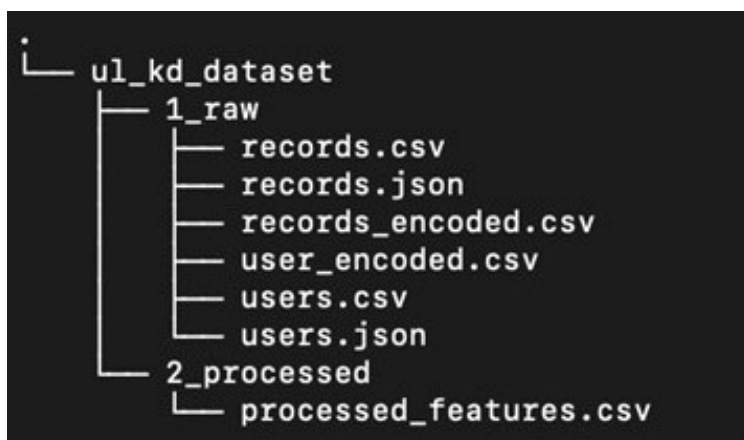
The dataset contains 10 994 keystroke patterns captured between October 2nd to November 17th, 2020. During the data collection, 66 participants were registered, with 59 % being male and 41 % female. The participants' ages ranged from 19 to 23 years old, situating the population within the young adult generation.

The dataset files are stored in Mendeley Data cloud-based communal repository and are organized hierarchically within the main folder, "ul_kd_dataset," as illustrated

in Figure 1. This folder contains two subfolders, namely “1_raw” and “2_processed”. The former includes raw data exported directly from the MongoDB Atlas database used by the web application, while the latter consists of processed data derived from raw data. The “1_raw” subfolder contains two main files: “users” and “records.” The “users” file includes data of the participants registered for the experiments, who signed up and completed a form on the web portal. The “records” file encompasses all keystroke patterns produced by the participants. Within the “2_processed” subfolder, there is a file containing time vectors (referred to as features), which can be directly used for training and testing classification or authentication models. Each entity mentioned above is available in both CSV and JSON formats. A more detailed description of the content of each file is presented below.

Figure 1

Dataset Structure



The “users” file, as mentioned before, contains information related to the participants, including personal data, typing style requested during registration, as well as certain metadata captured during the account creation. Table 1 presents the file fields in detail.

Table 1

Structure of the users_encoded.csv File

Field	Description
_id	System-generated unique user identifier.
name	User’s name.
lastname	User’s last name.

(continues)

(continued)

Field	Description
age	User's age.
email	User's email.
username	Username chosen by the user to perform the login tests on the system.
password	Password chosen by the user to perform the login tests on the system.
dni	National identity card.
isImposedPassword	Boolean value indicating whether a fixed password was imposed at the time of registration. "True" indicates a fixed password; "False" indicates a variable password.
genre	User's gender.
handedness	User's dominant hand.
handDisease	Boolean value indicating if the user is experiencing motor issues with their hands.
date	User's registration date in the system.
ipAddress	User's IP address at the time they registered in the system.
userAgent	Browser agent used to log into the system.

The "records" file includes keystroke samples made by users during the conducted experiments. Each pressed and released key was recorded in the dataset along with the associated timestamp at the time the participants entered their "username" and "password". Additionally, metadata of data collection sessions was stored, providing context for the tests conducted by each participant. Table 2 presents the details of each field in the file.

Table 2

Structure of the records_encoded.csv File

Field	Description
_id	System-generated unique record identifier.
rawUsernameKeydown	A collection of keystroke events recorded during the input of the username, specifically when the key was pressed, including both the pressed key and the corresponding timestamp.
rawUsernameKeyup	A collection of keystroke events recorded during the input of the username, specifically when the key was released, including both the released key and the corresponding timestamp.
rawPasswordKeydown	A collection of keystroke events recorded during the input of the password, specifically when the key was pressed, including both the pressed key and the corresponding timestamp.
rawPasswordKeyup	A collection of keystroke events recorded during the input of the password, specifically when the key was released, including both the released key and the corresponding timestamp.

(continues)

(continued)

Field	Description
belongedUserId	Identity of the user who created the credentials displayed in the session.
performedUserId	Identity of the user who owns the records of the current sample.
date	Date the sample record was created.
sessionIndex	The user must perform three authentication cycles per session. This integer value indicates the index.
valid	Users could make mistakes when typing. This Boolean value indicates whether the user correctly typed the credentials displayed on the screen the first time.
username	Username of the associated record.
password	Password of the associated record.
ipAddress	User's IP address at the time they registered in the system.
userAgent	Browser agent used to log into the system.
token	Unique token generated within the user's browser the first time they perform the login tests. This token can be used to detect if the same computer is used by the user to perform the tests.

In the "processed_features" file, features are generated from the difference between the captured and stored keystroke events and the raw data. Four time vectors were generated, namely ppTime, rrTime, prTime, and rpTime. Additionally, a final vector, which is a concatenation of these previous ones, is included. These vectors can be used to train and develop keystroke recognition models. Table 3 presents a detailed structure of the file.

Table 3*Structure of the processed_features.csv File*

Field	Description
_id	System-generated unique processed_features identifier.
belongedUserId	Identity of the user who created the credentials displayed in the session.
performedUserId	Identity of the user who owns the records of the current sample.
valid	Users could make mistakes when typing. This Boolean value indicates whether the user correctly typed the credentials displayed on the screen the first time.
password	Password of the associated record.
ppTime	Time vector generated by the difference in timestamps between two sequentially pressed keys.
rrTime	Time vector generated by the difference in timestamps between two sequentially released keys.
prTime	Time vector generated by the difference in timestamps between one pressed key and one released key.

(continues)

(continued)

Field	Description
rpTime	Time vector generated by the difference in timestamps between one released key and one pressed key.
vector	Concatenation of the four vectors (ppTime + rrTime + prTime + rpTime).
passLen	Length of the password.
vectorLen	Length of the vector created.

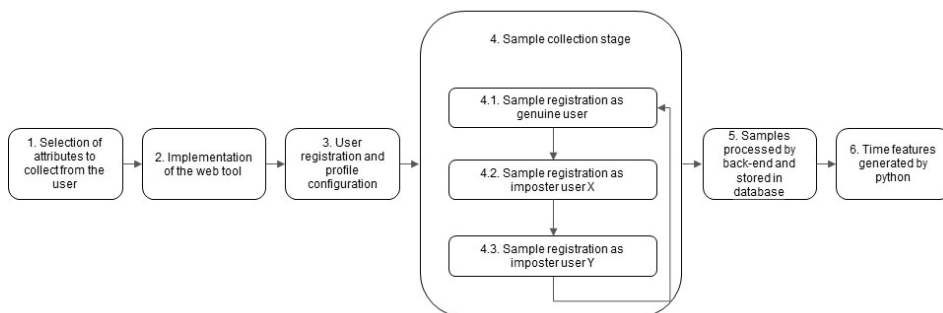
6. EXPERIMENTAL DESIGN, MATERIALS AND METHODS

6.1 Experimental Design

The process to generate the dataset of keystroke samples is detailed in Figure 2. The first phase involved implementing the web tool from scratch and identifying all attributes that could be collected from users and that would be relevant to keystroke dynamics models. Once these attributes were mapped, the second phase involved building a system with a front-end and back-end to enable the collection of these attributes. In the third phase, users who wished to participate voluntarily in the experiment were recruited and instructed to register on the web portal by filling out a form. After registration, users were required to access the web page on alternate days and complete a series of login tests displayed on the screen throughout the session. During these tests, every key pressed and released was captured as they entered the username and password. At the end of the session, all captures, which were stored locally, were sent to the service for subsequent storage. The collection phase lasted from October 2nd to November 17th, 2020, involving 66 users and recording a total of 10 994 records. The final stage involved generating the features that could be used in keystroke dynamics models.

Figure 2

Methodology Design

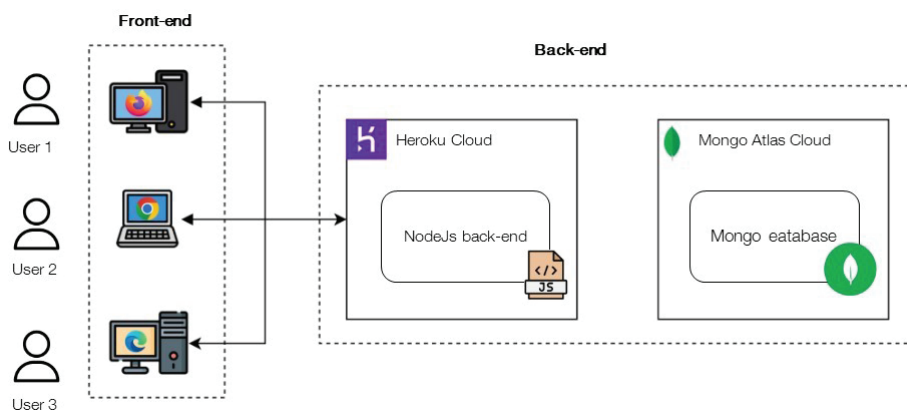


6.2 Materials

The collection system was implemented using web-based technology. The built solution, which includes both the front-end and back-end, is shown Figure 3.

Figure 3

Architecture of the Web Tool



The front-end, the visual layer where the end user interacts with the platform via a browser, was implemented using templates developed with the PUG library (version 3.0.0), JavaScript, and CSS. All screens incorporate these three components and are rendered on the back-end, then sent to the browser each time the user accesses the website URL. The logic for capturing keystrokes in the browser was written in JavaScript, the structural aspects of the web page were handled in HTML/PUG, and the visual design was crafted with CSS.

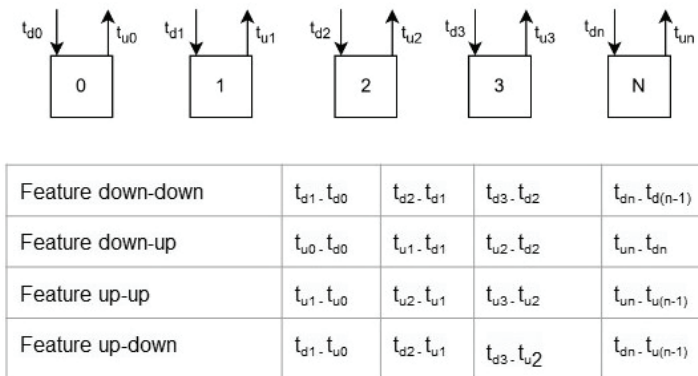
The back-end was developed in JavaScript using the Express library (version 4.17.1) and deployed on the Heroku cloud platform in a Node.js runtime environment. Different endpoints were created for each screen outlined in the user flow designed for keystroke capture sessions. When a user completes the capture flow, the front-end sends all records to the back-end, where they are received, processed, and stored in the database. Interaction with the non-relational MongoDB Atlas database was facilitated by the Mongoose library (version 5.10.5). Additionally, an email service using Mailgun was implemented to notify users of the proper storage of their session data.

Due to the unstructured nature of the keystroke capture records, a document-oriented non-relational database was chosen for storage. The MongoDB Atlas cloud was used to deploy the MongoDB Atlas database. Data was exported to monitor progress and maintain backups. The document structure used in the database is described in the Data Description section. The code used for the solution is provided in Appendix 6.

After the sampling period, various features were generated to form time vectors used for model training, including: the “down-down key feature,” “down-up key feature,” “up-down key feature,” “up-up key feature,” “hold time,” and “total time.” The first four features involve latency between different pressed keys, either when they are pressed (“down”) or released (“up”). “Hold time” refers to the duration a single key is pressed, while “total time” encompasses the entire time taken to type a word. Classification models cannot directly process the raw data generated by the tool as it only captures the timestamp when a key is pressed and released. To make this dataset usable, the raw data is processed to generate the aforementioned features. Figure 4 illustrates the generation of these time vectors, capturing both the “down” (td) and “up” (tu) events for each key. The latencies or differences between these events allow for the construction of user-specific vectors that will subsequently be used in the models. These features were generated using a Python script, which is also available in the GitHub repository mentioned in the Appendix 7.

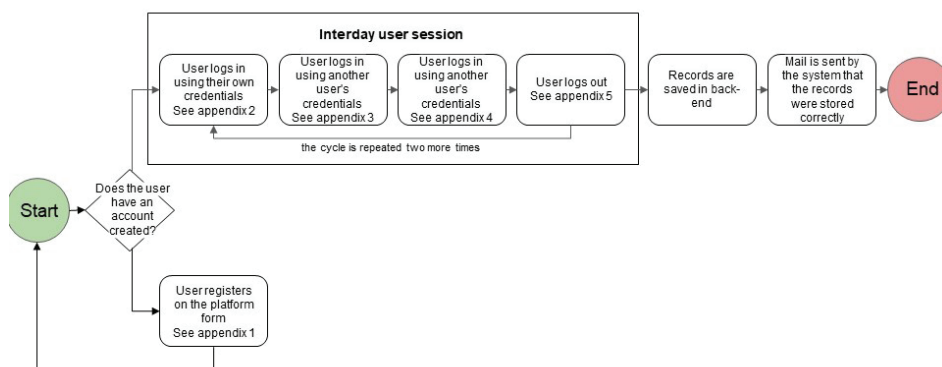
Figure 4

Procedure to Generate Time-Based Latency Features



6.3 Methods

Figure 5 shows the interaction flow proposed during the experimental design phase. The web tool was implemented to ensure that users could follow the proposed flow. The complete screen flow is detailed in the appendices.

Figure 5*User Interaction Flowchart*

Each week, users were required to complete at least two sessions, spaced on alternate days, to prevent “over-familiarity” with typing patterns while still allowing for user variability to be recorded. Each session consisted of three tasks, which the user performed three times, as depicted in Figure 5. In the first task, the legitimate user logged in to the system with their own username and password (either self-selected or assigned, depending on the group). Subsequently, they proceeded to a second and third screen where they typed the username and password of another randomly selected user three times. Once this flow was completed, the user logged out and repeated the process two more times. It is worth mentioning that for each entry to be considered valid, the user had to input the credentials correctly on the first attempt, with no corrections allowed, such as using the backspace key to retype. The session ended when the tool recorded the required number of correct samples according to the proposed methodology. Regarding invalid samples, they were also stored as part of the dataset, as the errors could be considered features related to the user’s identity and could be useful in future analyses. The data collection period lasted six weeks; however, the tool continued to capture samples beyond this period.

7. LIMITATIONS

In this experiment, part of the methodology involved sequentially typing the phrase that appeared on the screen three times and then repeating this process two more times. While this ensured more than 10 records per session, the experimental design, which included multiple writing sessions per user, may have introduced a bias in the data due to the repetitive nature of the tasks. This might lead to users learning and adjusting their writing patterns as the experiment progresses.

8. ETHICS STATEMENT

Following the ethical publishing guidelines provided by Elsevier and *Data in Brief*, the following key ethical aspects were considered:

Human Studies: All users participated voluntarily in the experiment and were informed about both the session dynamics and the intended use of the collected data. They provided a consent by filling out a form.

9. DATA AVAILABILITY

Data resources can be found in the following link: <https://data.mendeley.com/preview/9cg3c8jkh8?a=f4e49f3e-b689-4b6c-95a3-3bf5207d1935>

10. CREDIT AUTHOR STATEMENT

Aron Lo Li was responsible for the conceptualization and execution of the experiments. Juan Gutiérrez-Cárdenas played an active advisory role across all project stages. Victor H. Ayma assisted in shaping and enhancing the proposed methodology.

11. ACKNOWLEDGEMENTS

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. We thank the editor, reviewers, and all participants for their involvement and contributions to this research endeavor.

12. DECLARATION OF COMPETING INTEREST

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

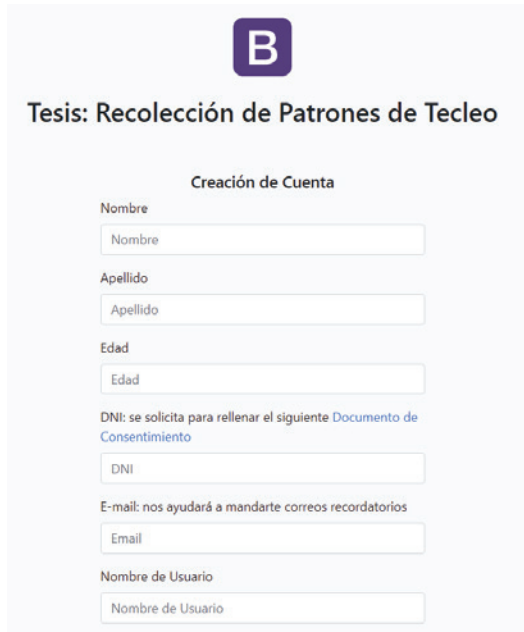
13. REFERENCES

- Giot, R., El-Abed, M., & Rosenberger, C. (2009). GREYC keystroke: A benchmark for keystroke dynamics biometric systems. In *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems (BTAS 2009)*, Washington D.C., United States, 1–6. <https://doi.org/10.1109/BTAS.2009.5339051>
- Killourhy, K. S., & Maxion, R. A. (2009). Comparing anomaly-detection algorithms for keystroke dynamics. In *2009 IEEE/IFIP International Conference on Dependable Systems & Networks*, Lisbon, Portugal, 125–134. <https://doi.org/10.1109/DSN.2009.5270346>

14. APPENDICES

Appendix 1

User Account Creation Screen



The screenshot shows a web form for account creation. At the top is a purple square logo with a white letter 'B'. Below it is the title 'Tesis: Recolección de Patrones de Tecleo'. The form is titled 'Creación de Cuenta' and contains several input fields: 'Nombre', 'Apellido', 'Edad', 'DNI' (with a note: 'DNI: se solicita para rellenar el siguiente Documento de Consentimiento'), 'E-mail: nos ayudará a mandarte correos recordatorios', and 'Nombre de Usuario'. Each field is a simple white box with a light gray border.

Appendix 2

Legitimate User Login Screen (Task 1)



The screenshot shows a web form for user login. At the top is a purple square logo with a white letter 'B'. Below it is the title 'Tesis: Recolección de Patrones de Tecleo'. The form is titled 'Tarea 1: Inicio de sesión'. Below the title is a paragraph of text: 'En esta tarea, usarás el nombre de usuario y la contraseña de registro para iniciar sesión. Si no te acuerdas, puedes verificar el correo que te mandamos con las credenciales de tu cuenta.' There are two input fields: the first contains the text 'aronlo98' and the second contains seven dots. Below the input fields are two blue buttons: 'Iniciar Sesión' and 'Registrarse'. At the bottom of the form is a line of text: 'Si tienes alguna duda, puedes contactarte al 959 291 344 / aron.lo.li@hotmail.com'.

Appendix 3

First Screen for Logging in as an Imposter User (Task 2)

B

Tesis: Recolección de Patrones de Tecleo

Bienvenido Aron

Tarea 2: Ahora tendrás que hacerte pasar por un usuario impostor e ingresar las credenciales del siguiente usuario. NO lo uses para iniciar sesión en la pantalla inicial.

Usuario
aylin1234

Contraseña
aylin1234

Captura válidas recolectadas:
2
67%

Nombre de Usuario

Contraseña

Iniciar Sesión

Appendix 4

Second Screen for Logging in as an Imposter User (Task 3)

B

Tesis: Recolección de Patrones de Tecleo

Tarea 3: Así como antes, ahora tendrás que hacerte pasar por otro usuario impostor e ingresar las credenciales del siguiente usuario. NO lo uses para iniciar sesión en la pantalla inicial.

Usuario
aylin1234

Contraseña
aylin1234

Captura válidas recolectadas:
2
67%

Nombre de Usuario

Contraseña

Iniciar Sesión

Appendix 5

Third Screen for Logging in as an Imposter User (Task 3)



B

Tesis: Recolección de Patrones de Teclado

Tarea 3: Así como antes, ahora tendrás que hacerte pasar por otro usuario impostor e ingresar las credenciales del siguiente usuario. NO lo uses para iniciar sesión en la pantalla inicial.

Usuario
aylin1234

Contraseña
aylin1234

Captura válidas recolectadas:
3

100%

Nombre de Usuario

Contraseña

Cerrar Sesión

Appendix 6

Source Code of the Web Tool Implemented for Dataset Generation

<https://github.com/aronlo98/tesis-keylogger>

Appendix 7

Source Code of the Keystroke Dynamics Models

<https://github.com/aronlo98/tesis>

