National College of Ireland

A framework to prevent insider threats by mitigating lunchtime attacks using biometrics

MSc Research Project
Cyber Security

Goutham Tattur Nagaraja
Student ID: 19237243

School of Computing
National College of Ireland

Supervisor: Dr. Vanessa Ayala-Rivera

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Goutham Tattur Nagaraja		
Student ID:	19237243		
Programme:	MSc in Cyber Security	**Year:**	2021-2022
Module:	MSc Internship		
Supervisor:	Dr. Vanessa Ayala-Rivera		
Submission Due Date:	16/12/2021		
Project Title:	A framework to prevent insider threats by mitigating lunchtime attacks using biometrics		
Word Count:	5459	**Page Count**	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Goutham Tattur Nagaraja
Date:	15/12/2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

A framework to prevent insider threats by mitigating lunchtime attacks using biometrics

Goutham Tattur
Nagaraja 19237243

Abstract

Electronic devices are becoming increasingly crucial for essential activities such as e-commerce, e-voting, and e-banking as people rely more on them. Biometric authentication methods such as fingerprint verification and facial recognition are widely regarded as highly effective. In current systems, the user is only verified at the point of entry into the system and is not authenticated later. This may lead to unauthorized users gaining access to the system. This paper proposes that the user should be verified at frequent intervals as it becomes increasingly crucial to make sure the system is being accessed by an authorized user throughout. In this paper, we present a multi-authentication framework that combines fingerprinting verification and facial recognition as methods to authenticate authorized users. The proposal involves continuously verifying the biometrics of the user at regular intervals of time to be sure that the genuine user is accessing the system. The framework was built using Raspberry pi with python as the developing language with the help of a Pi camera and external fingerprint sensor. A dataset was used to train the raspberry pi using deep learning for accurate facial recognition and liveness detection resulting in capturing the user's biometrics efficiently.

1 Introduction

In the 21st century, the use of electronic systems, such as CCTV cameras, biometric readers for access control, or technology depending on the use of identification cards or various forms of authentication, is common for supporting security and monitoring (Mohammad Dastbaz, 2015). A few, impacting government organizations, increasing the budget that they must allocate to installing and maintaining these systems, increasing the number and cyber security officers who must be trained to use this technology effectively, etc. Furthermore, owing to technological restrictions and the information and records provided by the foregoing systems and equipment, they might not be able to satisfy many of the demands of the ultimate user (M Wilson, 2003). In terms of information system development, the field has seen a trend towards automation. AI and its repercussions have become more prevalent as a result of these advancements. This has enabled robust devices to be used with low consumption of resources (Mohammad Dastbaz, 2015).

The motivation to propose this framework lies in incidents of shoulder surfing stealing user data

and accessing the system without their presence. In the UK alone the cost of a data breach due to shoulder surfing is 2.53 million pounds (Business Tech, 2021). Although there are multifactor authentication, privacy filters, etc. in the system now. However, the data cost of breach records clearly states that the breaches are prevailing, and much harder techniques are required for user verification (IBM Security, 2020).

This study develops a more robust identification technique based on the Raspberry Pi, which is presently one of several technologies with the most applications to meet the aforementioned requirements. Due to the small size, great compatibility with many other hardware devices such as fingerprint sensors and facial recognition cameras, and the use of a stable and easy-to-use operating system, while at a reduced cost, the mobile platform is a suitable platform for the development of security software (Halfacree, 2020).

The facial recognition and fingerprint run on an algorithm that is the largest trend in security automation. It works in real-time by capturing faces and also static by recognizing previously captured images. Whenever an individual is verified at the entry to a facility or when perpetrators are being tracked, it is frequently utilized by law enforcement authorities to validate the innocent, guilty, or complicity of the individual being sought in illegal activities (Misael Fernando Perilla, 2017).

Authenticating the user is the first and most important step in preventing unauthorized access. To maintain the security of the system, verifying a user's identity is essential. The three most common methods of identification and authentication are used alone or in combination. A user's authentication is usually based on something they own (such as a key, magnetic card, or chip card) or know (such as a PIN or password, for example). Users are not identified in these traditional systems. Furthermore, the objects they use can be stolen, lost, forgotten, or disclosed. In general, passwords are often easy to access by colleagues and even occasional visitors. To access the system, biometrics is a much better method. The uniqueness of every person's biometrics can be a benefit in enhancing the security of the system.

A malicious insider is someone who benefits from their privileged access to data, systems, or users, and who takes actions that negatively impact the integrity, confidentiality, or availability of the user's information. Depending on the severity, this can be problematic for an individual or compromise the whole organization. The malicious user will use methods like shoulder surfing and stealing credentials to capture the password from the user and gain access to their account (Peng Foong Ho, 2014). Despite fingerprints being unique to each person, there have been incidents in which attackers have bypassed the authentication of a system (IBM Security, 2020). The malicious person gained access to the system when no legitimate user was

using it. A study conducted by the Cisco Talos security group indicates that fake fingerprints are effective at bypassing system authentication 80% of the time (Kundaliya, 2020). Therefore, the requirement of producing an entirely secure system cannot be met by just fingerprints alone. Face recognition is another option for identity verification. The advantage of this technique is that it is faster and less inconvenient for the user.

There are cyber-attacks called lunchtime attacks (Tyler Kaczmarek, 2018) in which the attacker attempts to break into the system when the user is out for lunch or not accessing the system. In the first scenario, the user forgets to log off after using the system, and in the second scenario, the malicious person sabotages the genuine user to gain access to the system.

1.1 Research Question

What is an effective method to mitigate against insider threats and prevent lunchtime attacks using biometrics?

Insider threats are caused by are malicious intent people who have legitimate access to the targeted users' assets, whether maliciously or unintentionally, to cause harm to the users' assets and personal information. In an organization insider threats are not necessarily current employees. It can also be from former employees, contractors, or partners who have access to an organization’s systems or sensitive information. These attacks pose a lot of damage due to the severity of taking over the system resources completely (Goldstein, 2020).

Lunchtime attacks are caused when the user is off for a short break leaving the system unlocked unintentionally. This leads to the attacker gaining access to the system. After initial authentication, users often spend long periods continuously using computing devices and services. During that time, the continuous presence of the originally authenticated user must be periodically re-affirmed, especially, in a shared workplace setting. Failure to do so can result in so-called Lunchtime Attacks. Such an attack occurs when a previously authenticated user walks away from her workplace, thus allowing the adversary to take over her login session and engage in potentially nefarious activity. This prompts the need for periodic re-authentication and/or continuous authentication. Hence, this paper proposes a technique to reauthenticate the user for every regular interval of time (Tyler Kaczmarek, 2018).

2 Related Work

This research focuses on integrating biometrics as an authentication method for continuous evaluation of users and safeguarding the system from insider threats while also considering the privacy and security concerns of users.

Raspberry Pi is used in a wide range of platforms that employ methods to identify people based on patterns and/or personal features (biometrics), with several aims that aren't all related to safety. While there is a lot of research involving facial recognition, voice, and fingerprints where scientific information is available, many of them were developed without a desire to obtain scientific information, and for which there is limited documentation. Listed are a few research that makes a significant contribution to the development of security applications using at least one of the topics covered in this research. The development of a system for pattern recognition of the subcutaneous veins on a person's back, the upper part - in the palm of their hand, with real-time recognition, based on a type of classification, developed by Joardar, Chatterjee, & Rakshit, was one of the projects with important results in this project. This project was undertaken to develop a software system using Palm Dorsa Subcutaneous Vein Pattern (PDSVP) as a biometric physical feature and to test the reliability and measurable characteristics of this pattern (Sandip Joardar, 2016). The prototype has been implemented on a low-cost computer board, this project utilizes the Raspberry Pi Model B +, an infrared-sensitive camera (the Raspberry Pi NoIR camera), and other electronic components to detect near-infrared radiation (Near Infrared Radiation - NIR). It is feasible to analyze the structure of blood vessels located in the inner skin layer in the palm and utilize this data to subsequently recognize people with the correct usage of these data. Sivaranjani & Sumathi's creation of a system for detecting characteristics in fingerprints and feet utilizing Raspberry Pi (Sivaranjani & Sumathi, 2015), particularly stands out among them, is one of those research connected explicitly to identifying persons using biometrics. The development of a trace extracting system is illustrated using a Raspberry Pi and Debian Linux adapted for ARM (Raspbian) architecture. The authors used CMake, g++, and Makefiles to apply OpenCV2.4.9 (a set of Open Source modules) to picture identification for the evaluation of biometric parameters. The researchers have used a fingerprint extraction algorithm called "Minutiae" for fingerprint recognition. However, the project does not address wear and tear and other algorithms that might be more relevant.

Using a Raspberry Pi framework, Vinayak Bharadi, Dhvani Shah, V. J Kaul, and Sameer Amrutia propose a low-cost approach to calculate and capture a person's fingerprint in "IoT based biometrics implementation on Raspberry Pi". “Reliable Identity Management System Using Raspberry Pi” clearly defines the feasibility of building a biometric system using Python libraries and MySQL for storage (Dhvani. K. Shah, 2015). In Fingerprint Authentication using Raspberry Pi based on IoT, a detailed description is given on how to calculate bifurcation, ridge endings, and match the points for fingerprint authentication. This report explains how to utilize

the camera setup on the Raspberry Pi to record the individual's face and how to use OpenCV to report the accuracy of face detection and characterization on the Raspberry Pi. This proposal is in line with what is proposed in the research.

In the journal published by the Institute of Electronics and Computer Science, Lativa titled Face recognition system on Raspberry Pi provides a detailed report on embedding facial recognition systems to the Raspberry pi framework. This paper gives the motivation to deploy the prototype using Haar feature-based cascade classifier and compares the best approaches to deploy facial recognition in raspberry pi (Olegs Nikisins, 2015).

The freedom to choose among various algorithms to deploy the biometric features in the prototype is gained from the article written by (Misael Fernando Perilla, 2017), this journal article summarizes the work and the results obtained from a variety of algorithms that are used for biometrics in security applications. This paper concludes by selecting the most efficient and effective algorithm that can be chosen to build a security application in Raspberry pi.

It is important to know the areas that Raspberry pi can be used to include more features in the system. The diversity of this device can be found in a journal presented by (Anand Nayyar, 2015). The various application that the hardware device can be stretched to achieve maximum potential can be gained by understanding this journal. This has helped in cross-checking various other features that can be included in the prototype (Anand Nayyar, 2015).

“Fingerprint Identification in Biometric Security Systems” is a journal article published that addresses the challenge of choosing the best fingerprint matching technique in an attempt to construct a solution that meets the needed efficiency and quality standards. The information of the algorithm for the prototype is extracted from this paper (R.Mary Lourde, 2010).

To accomplish real-time identification and recognition, viable fingerprint identification systems necessitate few mistakes and a quick computation time. Information regarding the processing of fingerprints using the ridges and valleys of the human finger is used in the prototype by gaining information from this paper written by M.S.Alama M. Akhteruzzamana A.K.Cherrrib (M.S. Alam, 2004).

The prototype's primary element is reading fingerprints with Artificial Neural Networks to give an effective similarity measure for biometric verification utilizing a fingerprint reader. The paper published by Hamsa A. Abdullah titled Fingerprint Identification System Using Neural Networks provides detailed information on the algorithm that can be implemented using the latest technology of artificial intelligence (Abdullah, 2012).

In “A Review of Facial Biometrics Security for Smart Devices” provides the scientific background on facial recognition becoming the trusted and user-friendly method for

authentication of users in devices that considers Confidentiality, Integrity, and Availability of an electronic device (Mary Grace Galterio, 2018).

The survey of the biometrics security system by Chien Le explains the main differences between the methods of biometric technology used to verify user identities and explaining about the advantages and disadvantages of personal data security systems (Le, 2017).

It is important to know the challenges while building a system that deals with the biometric identities of individuals and the difficulties that can be encountered while dealing with the prototype. A journal article by Anil Jain, S. Pankanti and Ruud Bolle provides information regarding the same (Anil Jain, 2006).

3 Research Methodology

The research proposal of multiple authentications to prevent insider threats and lunchtime attacks is majorly performed by capturing the fingerprint and the facial recognition of the user which are performed in the following ways.

3.1 Stage 1: Fingerprint

Fingerprint registration and matching are the two phases of fingerprint scanning (the matching can be 1:1 or 1: N). While registering, the user must enter his or her fingerprints. The software will analyze the finger scans, create a digit blueprint derived from the findings of the analysis, and save the blueprint. Whenever a user inputs a finger using a sensor, the system creates a finger blueprint and compares it to the blueprints in the digit libraries. For 1:1 matching, the system compares the live finger blueprint to a blueprint specified in the Module; for 1: N matching, or finding, the system searches the entire finger libraries for the matched finger. In both cases, the system will deliver the same outcome, whether successful or failed.

3.2 Stage 2: Facial Recognition

The facial recognition module implementation in the prototype is carried out considering the following steps:

3.2.1 Step 1: Gathering the Dataset

The initial public dataset is the primary step in creating a deep learning network. The photographs themselves, and the annotations that go with each photo, are required. These descriptions must be drawn from a limited number of options. In addition, the number of

photographs in every genre should be somewhat consistent (i.e., the same number of examples per category). If there are double as many identical photos as there are separate photos, and 5 times as many various pictures as there are identical images, the classifiers will be inherently skewed toward overfitting into these widely represented categories.

In machine learning, class imbalance is a common issue that may be solved in a variety of methods.

3.2.2 Step 2: Splitting the Dataset

The dataset is then split into two categories

1. A series of training
2. A collection of tests

The classifier uses a training data set to "learn" what each category appears like by generating assumptions on the input data and then correcting itself until assumptions are incorrect. The test dataset is used to evaluate the classifier once it has been trained.

It is critical that the training and testing sets are distinct and do not coincide. If the testing set is included in the training data, the classifier has an unjustified benefit because it has previously observed and "learned" from the testing cases. As a result, the test dataset is maintained distinct from the training dataset and utilized solely for network evaluation.

3.2.3 Step 3: Training the Network

The network is presently being trained using the training set of pictures. The objective is to know how to identify each of the sections in our labeled data. When the program makes a mistake, it learns from it and continues to improve. In most cases, this is accomplished using a gradient descent algorithm.

3.2.4 Step 4: Evaluation

It's time to put the trained network to the test. Each of the photos in the testing set must be shown to the network, which then will determine what the image's label is. Finally, these model predictions are compared to the ground-truth labels from the testing set. The ground-truth labels represent what the image category is. From there, the computation of the number of predictions classifier got correct and compute aggregate reports such as precision, recall, and f-measure, which are used to quantify the performance of the network.

This proposed framework uses raspberry pi as a device to detect the biometrics of an individual iterating for every specified minute to assess if the genuine user is using the system.

The ASCII code of a given key on the keypad would be allocated to the fingerprint reader in

this setup. The key picked would be determined on a person's most frequently used key, as determined by the sort of activity he performs on the system. According to research on cognitive behavior and computer interaction, the spacebar is by far the most frequently pressed key on the keyboard. The decimal number 32, or the binary number 0010 00002, is the ASCII value of the spacebar. The fingerprint reader will be given this value. As a result, based on this information and the user's requirements and use, it's critical to allocate the ASCII value of the keypad to the fingerprint reader for the most frequently used button. A true incident may be utilized to properly illustrate this point. Consider a scientist who is using the system and frequently presses the space bar. As a result, the fingerprint reader will be positioned on the spacebar, and fingerprint data will be recorded and confirmed each moment the scientist presses the spacebar. Using the most recently collected data, a timer of "n" minutes is set, and when the timer expires, the computer automatically locks and waits for the user to press the key again.

During the user login process, the verification step verifies a person's identity. While static approaches offer greater reliable user authentication than simple passwords, it's indeed impossible to track a user switch following login identification. On the other perspective, constant identification keeps a record of the user's fingerprints throughout the interaction. During the constant cycle, the user is monitored regularly while typing on the keypad, permitting for actual analysis. It means that a user's fingerprints are constantly checked after the initial verification, and if they don't match the user's profile, access to the system is prevented.

Mitigating Security issues as the suggested study aids in mitigating the security vulnerabilities that plague today's systems. Dictionary attacks, Shoulder Surfing, Guessing, Spyware, Brute Force, Social Engineering are some of the challenges that developers and users face today while trying to authenticate users.

Shoulder surfing is an easy method of obtaining a user's password by seeing their typing rhythm while logging in. The authentication method is not threatened by shoulder surfing. In the proposed study, even if the intruder executes shoulder surfing and obtains the credentials, he is unable to get total access to the system since the fingerprint reader captures the authorized person's fingerprints every two minutes. The scanner verifies the identity that has been saved.

Spyware is software that collects data from users. An attacker can inject spyware into a user's profile to obtain access to the person's assets. However, the suggested study can alleviate this because the computer locks itself after 2 minutes if it detects that a valid user is not logging in.

The process of obtaining sensitive knowledge by influencing users is known as social engineering. An intruder can only obtain the user's credentials through social engineering, but since no two individuals on the planet have identical biometrics, social engineering attacks are impossible since the system constantly checks the user's fingerprints.

In a brute force assault, the attacker attempts every possible sequence to guess the password. To make things easier, he employs methods that do this operation. The suggested research, on the other hand, has the potential to reduce the danger. Even though the intruder cracks the login details, he will be unable to crack the biometrics or the fingerprint reader. As a result, the system is safe.

4 Design Specification

The Design of the research proposal is as follows:

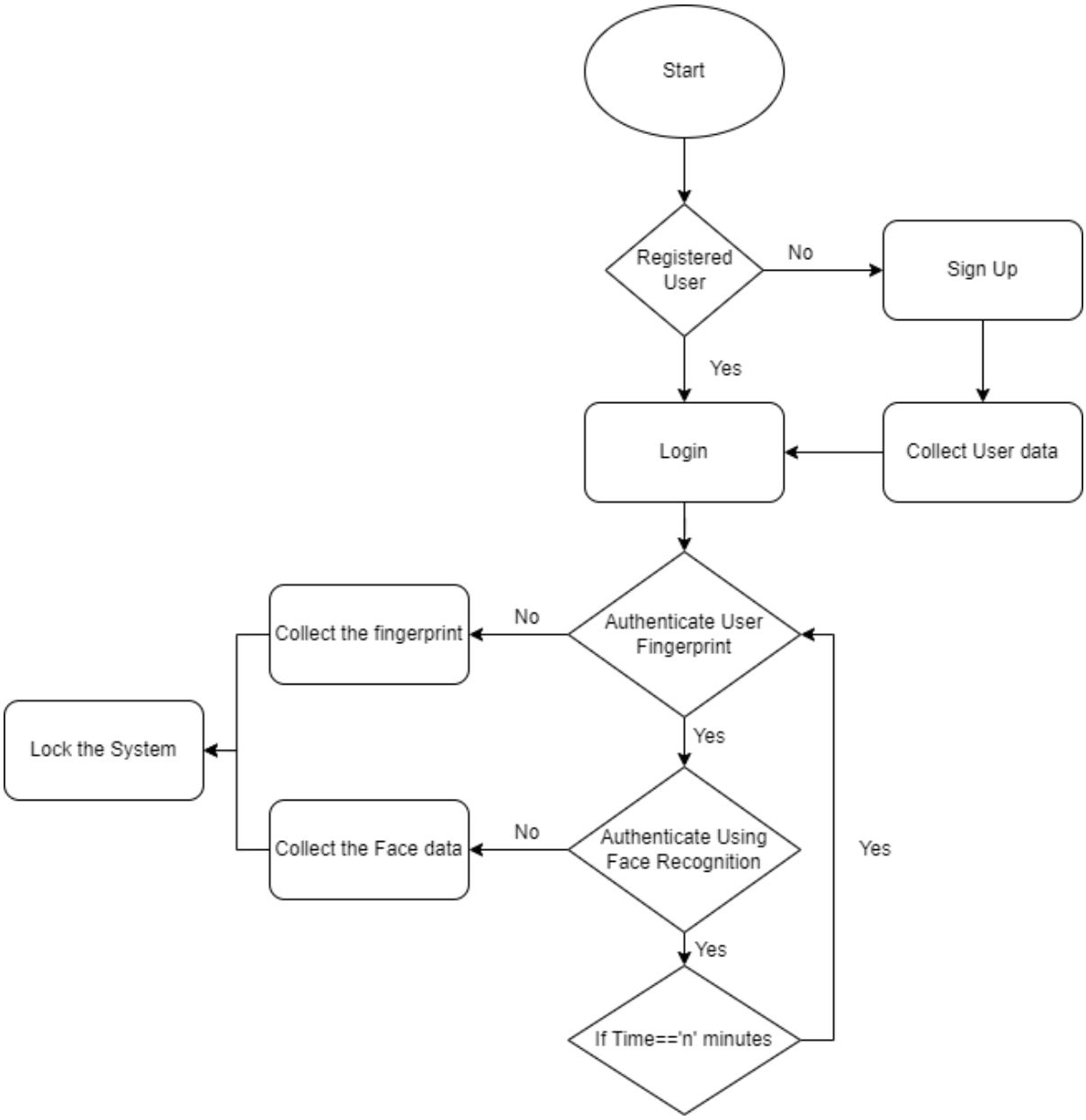

Figure 1 Flowchart of the proposed model

The suggested authentication method is depicted in the flowchart in figure 1. When the machine is turned on, the fingerprint recognition software is instantly started. The user must first signup by submitting his fingerprints and facial recognition information. The logged-in user must verify his or her identity with a fingerprint. The user then utilizes the system, and the fingerprint reader records the user's biometric data for the set "n" mins without his awareness. And the computer

locks itself if the user does not utilize a certain keyboard key (for example, Spacebar) in which the fingerprint reader is located. This prohibits anybody other than authorized users from accessing the system. There have been situations when a valid user's fingerprints have been collected and the user has been sabotaged, allowing a fraudulent user to take complete control over the system. As a result, verifying that the genuine user is using the system is indeed critical. This contributes to the system's security. The design of the proposed work is as follows:

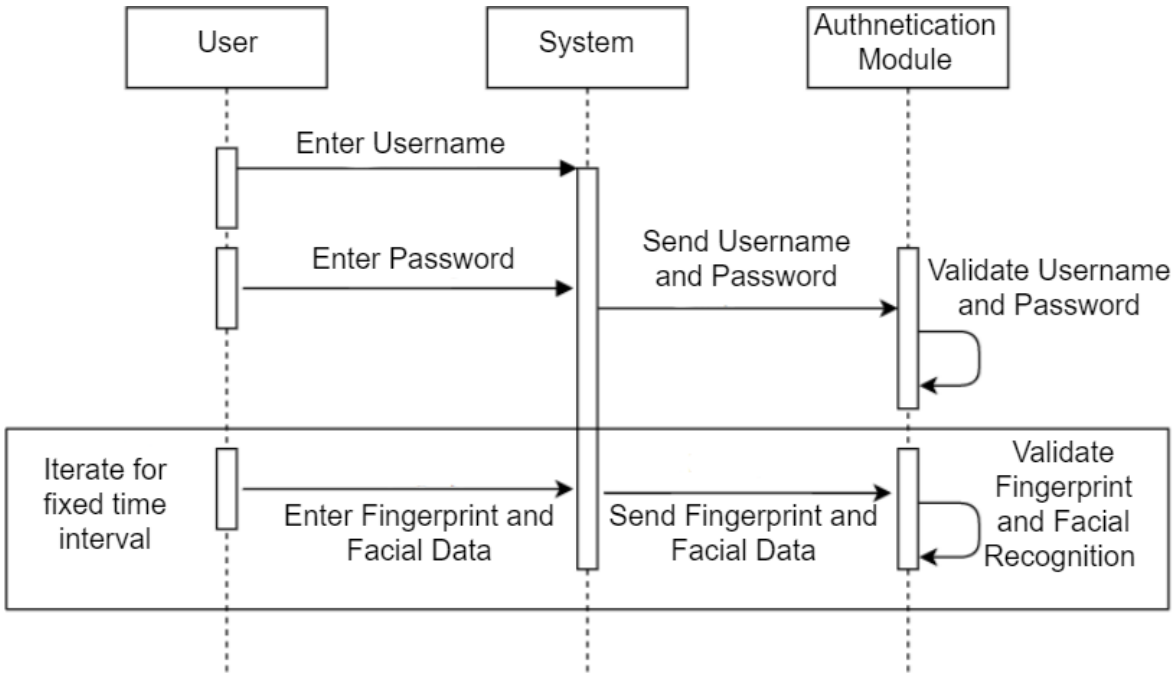

Figure 2 Sequence Diagram of the proposed model

The interactions amongst the many modules and characters of the system are depicted in figure 2. To begin, whenever a person logs into the computer, the system verifies his identity (what he knows). It looks for credentials. The system produces an alert and locks itself if the credentials are wrong. If the login and password supplied are accurate, the computer authorizes the user to proceed to the next phase of the verification procedure, which is fingerprint recognition. The user is then asked to verify his fingerprint on the fingerprint reader, which is located on the keyboard as it is the most frequently used key. The system will lock itself again if the user's fingerprint does not match. If the fingerprint is valid, the system grants the user authorization to utilize the system's resources. The fingerprint scanning function initiates a timer once the user login into the system and begins utilizing the resources, and it checks that the authorized user is using the system every "n" minutes. If the countdown has expired and

the user has not yet touched on the fingerprint scanner key, the system will remind the user to do so, and the user will be confirmed once again. If the fingerprint is failed or the user is using an application that does not involve the keyboard, then the system captures the image of the user and uses a facial recognition module to validate if the legitimate user is using the system. This prevents the system from falling into the wrong hands.

5 Implementation

The implementation of the prototype is carried out by hardware using Raspberry pi, fingerprint sensor, and Pi camera which is detailed as below:

5.1 Hardware Design Specification

5.1.1 Raspberry pi 4 64-bit quad-core Cortex-A72 processor and 2GB RAM

Raspberry Pi OS is a Debian-based free operating system that is designed for the Raspberry Pi device. Over 35,000 packages are included with Raspberry Pi OS, which is precompiled software bundles in a convenient format for simple installation on the device. Raspberry Pi OS is a community-driven project that focuses on enhancing the stability and performance of as many Debian packages are appropriate. The need for fingerprint validation and facial recognition can be easily carried out using this device and coding it with python makes it much simpler and more convenient.

Figure 3 Raspberry Pi

5.1.2 Pi Cam

In this study, the Raspberry Pi camera module is employed to record the user's high-definition facial image. A variety of libraries are also employed to picture capture the face in low light.

A five-megapixel fixed-focus camera on the module captures video in 1080p30, 720p60, and VGA90 formats, as well as stills. It connects to the Raspberry Pi's CSI port via a 15cm ribbon wire. There are various third-party libraries for accessing the features that the Pi Cam offers.

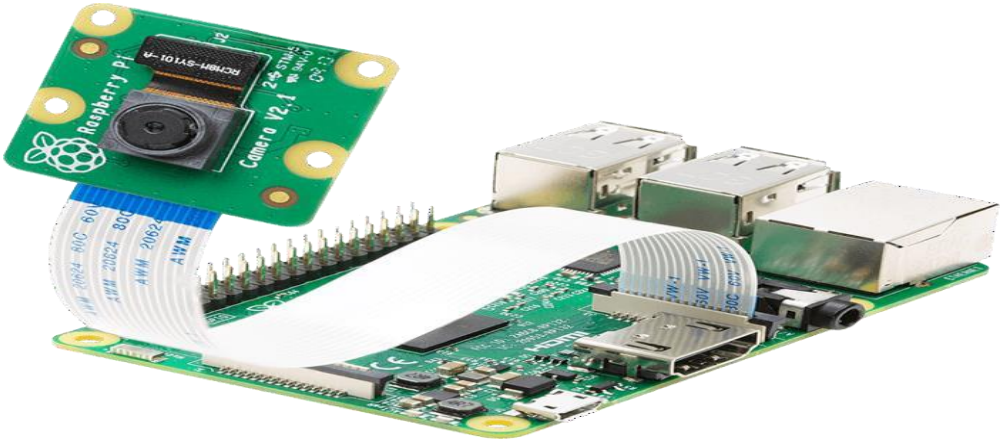

Figure 4 Raspberry Pi Camera

5.1.3 Fingerprint Scanner

As a biometric sensor, the fingerprint scanner – Time To Live (TTL) (GT-521F32), chosen due to its small and compact size, was used for system implementation.

Scan, extraction, pattern development, and comparison are the four processes involved in capturing the fingerprint of the user. The optical reader is the most popular fingerprint scanner. It works by reflecting light onto complementary metal oxide semiconductor sensors, which subsequently transform the light into electrical impulses. Fingerprint grooves and valleys produce various electric currents in different parts of the finger. Each of its finger's ridges and valleys acts as an insulator, limiting the flow of current generated by the reader.

Once these data are saved, the identification of the legitimate user fingerprint is scanned and verified. The scanner frequently requests numerous prints from the same finger to construct a comprehensive and accurate fingerprint picture of the user's fingerprint (Joao Gaspar, 2019).

Figure 5 Fingerprint Sensor

5.2 Software Design Specification

5.2.1 Python 3.7.4

This complete research prototype is built using python. The reason is it is user-friendly, and a large number of available libraries allow us to fulfill the requirements of the prototype in the development. The libraries that are used are as specified in table 1.

Serial	import serial
Tkinter	import tkinter
Imutils	from imutils.video import VideoStream
keras	From keras.preprocessing.image import img_to_array
Keras	from Keras.models import load_model
numpy	import numpy as np
imutils	import imutils
pickle	import pickle
Opencv	import cv2

Table 1 : Python libraries for the development

5.2.2 **Putty:** The Putty software is used for SSH connection between the raspberry pi and the user system. It is a Telnet for Windows and Unix platforms

5.2.3 **VNC viewer** is a graphical desktop sharing technology that lets users control the Raspberry pi interface from the host system.

5.2.4 **Advanced IP scanner**: The advanced IP scanner is a tool to analyze LAN and to recognize the IP address that the raspberry pi is allocated to.

6 Evaluation

The prototype for multiple authentications using fingerprint and facial recognition is successfully deployed and the matrices for Liveness detection, Accuracy of the recognition are considered to evaluate the prototype.

The liveness detection is used to verify if the real live user is accessing the system. If there is any image shown of the user the prototype immediately rejects the face and logs out of the system.

6.1 Experiment 1: Liveness Detection

The prototype is built to detect the liveness of the users depicted in figure 5. A fake image will not be detected. the system will provide access to the real users and any photographed image shown to the camera will be rejected. This helps in mitigating if the user is sabotaged by the attacker.

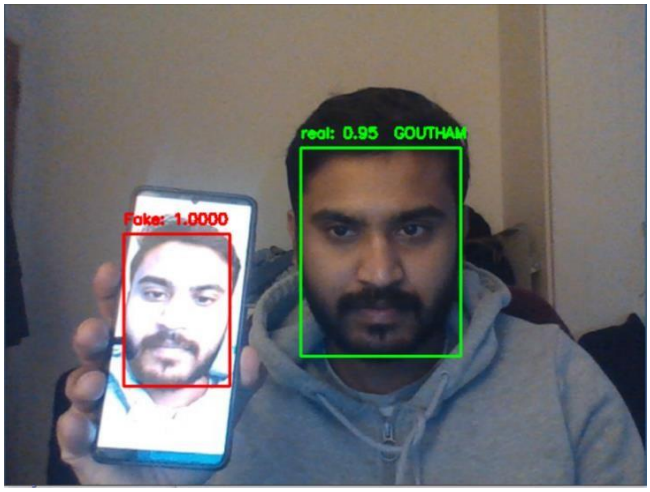

Figure 6 liveness Detection

6.2 Experiment 3: Accuracy of the Fingerprinting detection

Image figure 7 explains the verification of user fingerprints. The timer is for 5 seconds to retrieve the fingerprint and once the fingerprint is successful the system logs in.

Figure 7 Fingerprint Login Successful

If the fingerprint of the user is invalid, the system tries 3 times, and if it fails the control moves to capturing the facial recognition of the user as shown in figure 8.

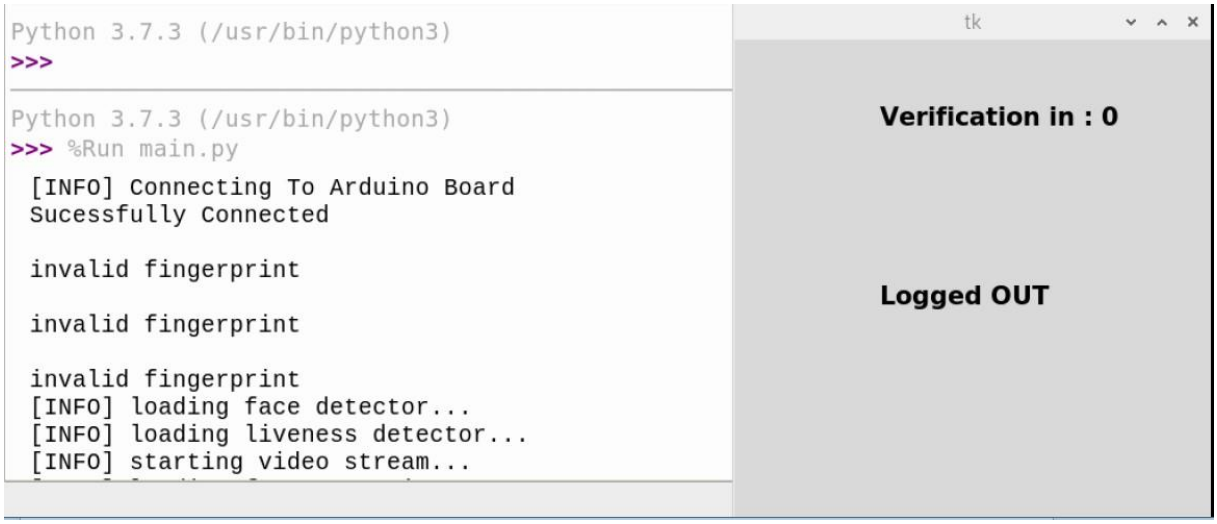

Figure 8 Fingerprint Failure verify face recognition

6.3 Experiment 2: Accuracy of the Facial recognition

Figure 9 displays the accuracy of the facial recognition and the loss on the validation and training dataset. The training and validation loss values have both been significantly lowered as the Epoch value i.e., as the number of images looped increases the training and validation loss is decreased. Contradicting, the Training accuracy value and the validation accuracy value are increased and is remained constant with 100% accuracy. Hence, observing this graph we can conclude that the model is trained to recognize the real face of the authenticated user.

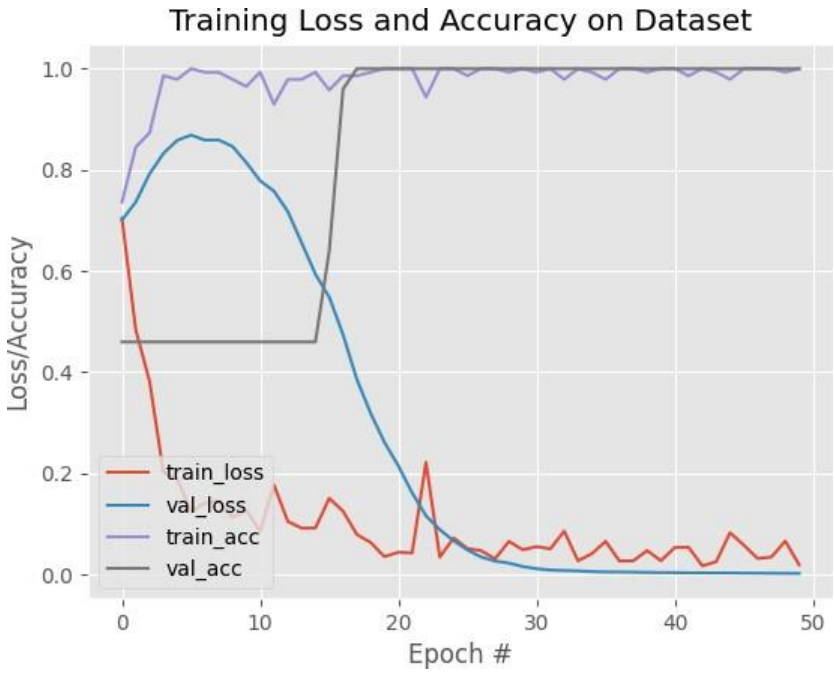

Figure 9 Face detection Training and accuracy

The facial recognition is captured by the user. It can be observed that the prototype can capture and verify if the legitimate user is using the system in low light conditions as shown in figure 10.

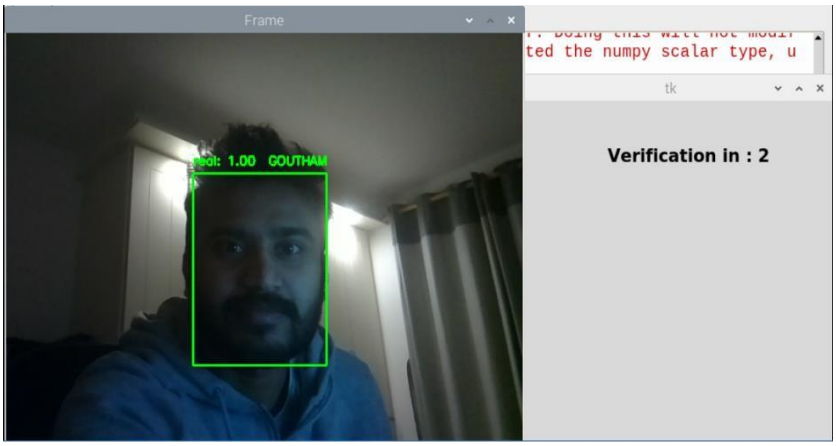

Figure 10 Facial recognition successful

7 Discussion

The proposed research enhances multiple authentications and verifies users constantly throughout the session. The previous biometric system does not involve constantly verifying the users and authenticating only at the entry point.

The limitation of the research is deploying this model without affecting the user interface and integrating these features in the upcoming devices.

This model can be integrated into various other applications like server rooms, security rooms, etc. The proposed research can also be enhanced in the future by including many other biometric features and creating a system where each action that the user makes with the system can be recorded and could be used for checking if the genuine user is using the system. Further development in this field can be diverse and make more secure systems in the future.

8 Conclusion and Future Work

This research work identifies the authorized user's identity in the system by asking the user to verify with their fingerprint and facial scan. The prototype successfully identifies the uses in every certain interval of time to ensure it’s the same authorized user of the system. This prototype will not only solve the issue of lunch-time attacks, but it will also resolve the issue of insider threats. This prototype also ensures that the facial scan of the user is of a real person and not a digital imprint of the user, this indicates there is a higher accuracy rate of verifying the identity and rejecting if there is a false match. In the future, this prototype can be induced inside the system itself, for example, the fingerprint scanner can be embedded on the spacebar as it is the most used key and will keep identifying the user in interval gaps. This system can also be further developed by adding more stronger identification methods, for example, an iris scanner which will scan the user’s iris as it is unique for every individual and similarly ensure the user is not impersonating to be the authorized user of the system. In conclusion, the proposed approach was successfully implemented and will be a great suit for highly secured firms.

References

Abdullah, H. A., 2012. Fingerprint Identification System Using Neural Networks. *Al-Nahrain Journal for Engineering Sciences,* 15(2), pp. 234-244.

Anand Nayyar, V. P., 2015. Raspberry Pi-A Small, Powerful, Cost Effective and Efficient Form Factor Computer: A Review. *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE),* Volume 5, pp. 7-9.

Anil Jain, R. B. S. P., 2006. Introduction to Biometrics. pp. 1-41.

Business Tech, 2021. *What is shoulder surfing?.* [Online]
Available at: https://www.businesstechweekly.com/cybersecurity/password-security/what-is-shoulder-surfing/
[Accessed 15 11 2021].

Dhvani. K. Shah, D. V. A. B. V. J. K. S. A., 2015. *End-to-end Encryption based Biometric SaaS.* Mumbai, International Conference on Computing Communication Control and Automation.

Goldstein, J., 2020. *What Are Insider Threats and How Can You Mitigate Them?.* [Online]
Available at: https://securityintelligence.com/posts/what-are-insider-threats-and-how-can-you-mitigate-them/
[Accessed 01 12 2021].

Halfacree, G., 2020. *The official Raspberry Pi beginner's guide: how to use your new computer*. 4th ed. Cambridge: Raspberry Pi Trading Ltd.

IBM Security, 2020. Cost of a Data Breach Report 2020. *IBM Security,* Volume I, p. 82.

Joao Gaspar, R. B. F. P. S. N. S. O. A. P., 2019. *Anti-UAV Mobile System with RTLS Integration and User Authentication.* Lisbon, Portugal, IEEE.

Kundaliya, D., 2020. *3D-printed 'fake fingerprints' can bypass fingerprint scanners, researchers warn.* [Online]
Available at: https://www.computing.co.uk/news/4013809/3d-printed-fake-fingerprints-bypass-fingerprint-scanners-researchers-warn
[Accessed 20 11 2021].

Le, C., 2017. A Survey of Biometrics Security Systems. *Washington University,* Volume 1, p. 3.

M Wilson, J. H., 2003. *Building an Information Technology Security Awareness and Training Program,* Gaithersburg, MD: s.n.

M.S. Alam, M. A. A. C., 2004. *Real-time fingerprint identification.* Kuwait, Elsevier.

Mary Grace Galterio, S. A. S. T. H., 2018. A Review of Facial Biometrics Security for Smart Devices. *Computers,* 7(3), p. 37.

Misael Fernando Perilla, B., 2017. Raspberry Pi for Biometrics and Security Applications. 7(8), p. 9.

Mohammad Dastbaz, S. W. H. E., 2015. Biometric Technology. In: S. Y. Babak Akhgar, ed. *Chapter 10 - Emerging Technologies and the Human Rights Challenge of Rapidly Expanding State Surveillance Capacities.* s.l.:Butterworth-Heinemann, pp. 108-118.

Olegs Nikisins, R. F. A. K. M. G., 2015. *Face recognition system on Raspberry Pi.* Latvia, Institute of Electronics and Computer Science, p. 9.

Peng Foong Ho, Y. H.-S. K. M. C. W. Y. N. C. L. Y. P., 2014. Preventing Shoulder-Surfing Attack with the Concept of Concealing the Password Objects' Information. *The Scientific World Journal,* Volume 2014, pp. 1-12.

R.Mary Lourde, D. K., 2010. Fingerprint Identification in Biometric SecuritySystems. *International Journal of Computer and Electrical Engineering,* Volume , pp. 8-20.

Sandip Joardar, A. C. A. R., 2016. Real-time NIR imaging of Palm Dorsa subcutaneous vein pattern based biometrics: An SRC based approach. *IEEE Instrumentation & Measurement Magazine,* 19(2), pp. 13-19.

Tyler Kaczmarek, E. O. G. T., 2018. Assentication: User De-authentication and Lunchtime Attack Mitigation with Seated Posture Biometric. In: F. V. Bart Preneel, ed. *Applied Cryptography and Network Security.* Cham: Springer International Publishing, pp. 616-633.