MoLaBSS: Server-specific add-on biometric security layer model to enhance the usage of biometrics

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY-NC-SA 4.0

SUBMISSION DATE / POSTED DATE

09-04-2020 / 12-06-2020

CITATION

Singh, Bhanu Pratap; Singh, Nirvisha (2020): MoLaBSS: Server-specific add-on biometric security layer model to enhance the usage of biometrics. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.12101577.v3>

DOI

[10.36227/techrxiv.12101577.v3](https://doi.org/10.36227/techrxiv.12101577.v3)

Article

MoLaBSS: Server-Specific Add-On Biometric Security Layer Model to Enhance the Usage of Biometrics

Bhanu Singh ^{1,*} and Nirvisha Singh ²

¹ Information Technology and Services, Perrigo Plc., Allegan, MI 49010, USA

² Department of Neuroscience, Central Michigan University, Mt. Pleasant, MI 48859, USA; singh3n@cmich.edu

* Correspondence: bhanu.singh@perrigo.com or spbhanu201@gmail.com

Received: 18 April 2020; Accepted: 2 June 2020; Published: 8 June 2020

Abstract: With high-paced growth in biometrics, and its easy availability to capture various biometric features, it is emerging as one of the most valuable technologies for multifactor authentication to verify a user's identity, for data security. Organizations encourage their members to use biometrics, but they are hesitant to use them due to perceived security risks. Because of its low usage rate, many medium and small segment organizations find it unfeasible to deploy robust biometric systems. We propose a server-specific add-on biometric security layer model (MoLaBSS) to enhance confidence in the usage of biometrics. We tested this model via a biometric mobile app, and the survey showed a favorable response of 80%. The innovative mobile app was tested for its usability and got a score of more than 71%. For test tool reliability, we examined the equal error rate (EER) of the app and got a reasonably low score of 6%. The results show good potential of this framework to enhance users' confidence level in the usage of biometrics. Higher usage rates may make deployment of biometrics more cost-effective for many organizations to decrease their information security risk.

Keywords: biometrics; mobile applications; data security; privacy; cyber security

1. Introduction/Background

With the strong demand for biometrics-as-applications services (BaaS) in the cyber world, especially for usage with IoT devices, there come security vulnerabilities [1]. The IoT ecosystem is still not entirely trustworthy because it faces many challenges due to the characteristics of cyberspace, where threats from organized intelligence and criminal groups are far more challenging to defeat than individual hackers [2]. Therefore, the fast growth of biometrics in cyber-society, without some extra layers of security, may pose some big problems, especially for medium and small healthcare organizations [3]. The reason small and medium healthcare organizations are more vulnerable is because their customers and patients are more focused on their health issues than systems' security. Moreover, unlike banks and financial organizations, healthcare organizations lack large fraud detection and security teams [4]. This provides more opportunities for hackers to steal patient medical records, clinical trial data, and drug-development-related information, which may increase counterfeit drug trafficking and medical insurance fraud [5]. A large percentage of users use obvious user IDs and passwords for many of their online transactions [6,7]. It is much easier to steal someone's password and user ID (UID) and type in the date of birth (DOB) or pet's name on a keyboard than to spoof someone's fingerprint or any biometric information and use it effectively [8]. Therefore, large healthcare companies are encouraging their users to use biometric technologies for user authentication in m-Health (mobile health apps). However, because of the low usage rate, many small and medium healthcare companies are not finding it very cost-effective to deploy a very robust multifactor authentication

(MFA) application system. A large percentage of users refrain from online usage of biometrics for privacy, security, and data leak issues [9].

For the above reasons, large corporations, especially large banks, have been encouraging their customers to download their app and use biometric data such as fingerprints for MFA. Users can download the banking app from the bank's website after logging in and install on any of their mobile devices. They can use either login and password or the sensor for fingerprint or facial recognition, depending on the model of their mobile device. Users can get SMS and/or email alerts if they have agreed to and saved this in their user profile. However, some security concerns have been raised by many critics against the mass usage of such apps [10]. Many users do not choose the notification option in their user profile to avoid too many alert messages. Even if a user has saved notification option in their user profile and chosen to use fingerprint-based login; a hacker can spoof a user's fingerprint and perform fraudulent transactions [11]. Hackers can use man-in-the-middle (MiTM) techniques, even if SSL/TLS pinning is used in the app [12]. There is a higher possibility of fingerprint spoofing in a hospital environment, despite the data and privacy protection rules such as the Health Insurance Portability and Accountability Act (HIPAA), General Data Protection Regulation (GDPR), California Consumer Privacy Act (CCPA), etc. [13]. A hacker can use the pre-hash manipulations technique or install the banking app on any device, steal user IDs and passwords, enter the DOB or pet's name, and perform fraudulent transactions without the user knowing about it for a long time [14,15].

We propose a server-specific add-on biometric security layer model (MoLaBSS), to enhance confidence in the usage of biometrics. This model is based on the premise that certain applications are more critical to users and building confidence in the use of those applications is more important than providing a general level of extra security layer for many applications. There are apps deployed by large financial institutions that can send a notification to registered users if someone logs in to users' accounts from another device and performs activities. There are hardware-based fingerprint readers like that of "Windows Hello" and HP's USB fingerprint reader that let a user verify a Microsoft account or a non-Microsoft service that supports Fast Identity Online (FIDO) [16].

Why, then, do we need an "add-on" app for an extra layer of biometric security to test MoLaBSS? There are many reasons to develop the Bio-Guard app, some of which are mentioned below:

- We did not find a feasible software solution that could fit in our model on the server-side and mobile device side to test the MoLaBSS.
- Architecturally, it is different, in that it is a software-based and not a hardware-based extra layer of security.
- There is an extra cost associated with hardware replacement/upgrade. A hardware-based extra layer of security may fail if there are any wear-and-tear-related issues.
- Software-based security would be more scalable and flexible to adjust to the fast-changing security dynamics [17].
- This app would send "server ID" only to the verified users to complete the installation of the app; without that, users will not be able to access a particular application. In this, it is very targeted and server-specific and not a generic security layer.
- Most of the methods of user or device authentication emphasize one or a few cryptographic or biometric features. Multimodal biometric systems combine many biometric features to make the authentication more scalable [18], and this app is designed along multimodal lines. It can be used for any of the biometric features captured by smartphones.

To test MoLaBSS, we designed, developed, and tested a low-cost mobile app "Bio-Guard," to provide an extra layer of biometric security for the access of a web-based application via mobile devices. We then surveyed a reasonably large sample population for the success of this framework.

The mobile app solution framework is detailed in Section 3 (technology) and Section 4 (experiments). We have presented experimental outcomes, survey results, and analysis in Section 5. Section 6 presents implementation opportunities and challenges. Section 7 concludes with some future perspectives.

In the next section, Section 2, we discuss some of the relevant authentication modalities, types of communications used for authentication, and related factors which can impact these processes.

2. Related Works

The IoT ecosystem is open or semi-open to many types of users, including hackers and adversaries [19]. To improve security in the entire IoT chain, the focus of research work is shifting towards more robust user authentication. Aghili et al. proposed a lightweight three-factor authentication access control (LACO) [20] for preserving the security and privacy of e-health records during data transfer. This model can suit resource-constrained sensors in the IoT chain. However, with the falling cost of sensors, there are other ways also available to authenticate data and verify a user's identity. One of the most-often-mentioned classification schemes of authentication is based on something-you-know (SYK), i.e., a password, something-you-have (SYH), i.e., a smart-card, and something-you-are (SYA), i.e., biometric features such as retina print, as discussed by [21,22]. Ferrag et al. [23] presented authentication schemes of mobile devices in four categories: (a) factor-based, e.g., 2-factor and multifactor; (b) channel-based, i.e., seamless roaming [24]; (c) ID-based, i.e., remote user's authentication [25]; and (d) biometric-based.

As per Traore et al. [26], authentication can be divided into two categories: static and continuous. The static user authentication gets invoked once at the beginning of a communication session, e.g., to verify the user's password/PIN [27] or facial features [28]. The continuous authentication gets repeatedly invoked to verify the legitimacy of a user based on touch or mouse movement/click or keystroke during the time of device usage [29]. Unfortunately, most of the smartphone authentication methods follow the static authentication process, which cannot provide seamless certification. Therefore, static authentication becomes prey to session hijacking attacks [30]. To overcome the issue of static authentication of biometric features and cryptographic methods, Robertas et al. [31] proposed an EEG-based biometric cryptosystem. This biometric cryptosystem has low explosibility and can be used by disabled people or users with some missing physical traits. For continuous authentication, Guannan et al. [32] suggested a two-step authentication method based on motion and physiological movement, i.e., photoplethysmography (PPG) signals, especially for wearable devices. Even continuous authentication protocols can be classified into two categories: (a) user-to-device and (b) device-to-device [33].

Based on a review of the above literature, it appears that the continuous authentication method mixed with biometric features and physiological movement can address many security issues of mobile devices and applications [34]. However, each of the biometric characteristics, such as fingerprint, iris, electroencephalography (EEG), voice, face, palm, ear, and gait data, has its advantage or disadvantage. For example, age has a minimal impact on features of the ear, and this can be a very convenient source of data for passive biometrics, but continuous authentication of the ear is difficult [35,36]. Robertas et al. [37] discussed gait-based biometric features, which can be unobtrusive, implicit, and passively observed continuously, so long as the user is carrying her/his smartphone or is around it, though gait can be affected by many factors such as shoes, stimulants, mood, aging, etc.

Hardware-based biometric security access technology of prominent vendors like HP and Microsoft are primarily designed for mass usage. There are many sensitive software applications, mainly related to patient health records, which would need more specific protection. To provide a different level of eHealth record access to different personnel of varying rank, Spanakis et al., proposed SpeechXRays, a multi-channel biometrics platform of user authentication [38,39]. This framework is based on a multimodal authentication paradigm, based on voice acoustics facial recognition [40,41]. The fundamentals of this approach are excellent; however, devices to collect relevant data and the willingness of users to share all these data at the moment could be a challenge.

Recently, mobile devices have started focusing on 3-D orientation of the biometric sensors. This has led to many research works in touch- and gesture-based continuous authentication, even though there are variations in the captured touch data, depending on the user's posture, despite the similarity in the

device specifications [42]. These continuous authentication techniques have inspired us to develop something novel, flexible, and scalable, and we developed a mobile app framework, Bio-Guard. This app framework uses a continuous authentication method based on session ID, which is in line with the latest schemes of lightweight authentication techniques recommended to be used in the era of IoT and IoS (Internet of Sensors) [43]. The adaptability of the Bio-Guard app distinguishes it from many other mobile app frameworks because it can adapt to any cryptographic or biometric features, used by the users via their mobile device. The architecture of the Bio-Guard app is open to be aligned with multimodal authentication and NGI systems. For criminal investigation and record access, more advanced “next-generation identification NGI” systems are suggested by many researchers. For example, NGI systems may authenticate many aspects of biometric features, such as tattoos, scars, etc. which can throw light on gang membership and previous convictions [44]. However, for day-to-day access to applications, NGI systems may be too cumbersome for ordinary users.

3. Technology and Methods

This section presents three subsections to provide a broad overview of the Bio-Guard app framework: 3.1. an overview of the architecture, technical design, and data flow; 3.2. the main functions of the app and its variables; 3.3. the authentication process of the Bio-Guard app.

3.1. An Overview of the Architecture, Technical Design, and Data Flow

Figure 1 represents an overview of how the data flows in a multi-user, multi-direction scenario, after the installation and registration of the Bio-Guard app.

Figure 1. An overview of the architectural diagram showing how biometric authentication data flows after installing the Bio-Guard App on a registered mobile device and Oracle biometric authentication server. The numbers 1a, 1b, 1c, 2 and 3, represent the multi-directional, multi-user sequential points. The processes are described in Figures 2a,b, 3a,b and 4a,b.

Points 2 and 3 are the two servers used in this process: the “authentication server with a biometric adapter” (AsBAA) and the “main host server” (MHS), as shown in Figure 1. The MHS has a web-based “Fingerprint-Based Attendance System” (FBAS) and server-side Bio-Guard app installed on it. In this scenario, MHS is protected by the company’s firewall, and the AsBAA is outside the firewall, but it could be different in the case of cloud services, or clustered architecture [45]. The Bio-Guard mobile client-side app is interfaced with a web app to push data to the MHS via AsBAA. The Bio-Guard mobile client-side app is installed on the user’s mobile device (points 1b and 1c), and the system administrator has access to the server-side app (point 1a), as shown in Figure 1. A user has an option to register from point 1b or 1c, using his/her regular Google/Facebook profile, as shown in Figure 1. The app gets a GPS location from the place where it is connected to the internet. The Bio-Guard app is based on Android studio code, an example of which is referenced at the link [46]. The web app was installed on Apache Tomcat with hibernate support and JBoss data services so that data to the central database (DB) could be updated via cookie-enabled sessions, using (representational state transfer) REST-ful web services. The app can allow users to log in with their Facebook or Google account, and via C-API (connector application programming interface) it can fetch their public fields, such as name, gender, user id, etc., provided such details are stored in the FBAS for registration verification, a code example of which is referenced at the link [47].

3.2. The Main Functions of the App and Its Variables

The installation of the server-side and client-side of the app is shown in Figure 2a,b, respectively. On installation of the server-side of the app, on the main host server (MHS), it generates “server installation ID” (SID), e.g., sid1234, and gets synched with the “authentication server with biometric adapter” (AsBAA), as shown in points 2 and 3 of Figure 1 and also in Figure 2a.

Figure 2. (a,b) demonstrate the generation of SID and client installation ID (CID) on the server and client-side of the Bio-Guard app, respectively, after the installation. (c) The Bio-Guard app after it expanded on a Samsung android mobile phone screen, after a single tap of the Bio-Guard icon.

On the installation of the client-side of the app on a mobile device, it generates client installation ID (CID), e.g., cid5678, and it also creates a Bio-Guard icon, as shown in Figure 2b and represented in points 1b and 1c of Figure 1.

A single tap of the Bio-Guard icon opens Bio-Guard on a mobile device for registration, as shown in Figure 2c. After the installation of the Bio-Guard app on a mobile device, the CID gets linked to the device id/international mobile equipment identity (IMEI) of the mobile device. Then, a user sends registration details to the MHS via AsBAA, as shown in Figure 3a. AsBAA sends the user's details to the MHS, which links the app's CID to the user profile and generates a trigger in the server side of the app to send an "one time password" (OTP) to the registered users to verify her/his identity. On verification of the user's biometric profile, the AsBAA sends OTP received from MHS to the user to verify his/her identity, as shown in Figure 3b. The user confirms their identity by confirming the received OTP by tapping the button "Confirm OTP to receive SID," as shown in Figure 2c. The verification of identity via the OPT confirmation generates another trigger in the MHS to send the SID to the AsBAA, which in turn sends SID to the registered user to complete the installation of the client-side of the app, as shown in Figure 3b. On a single tap of the submit button of the client-side of the app, as shown in Figure 2c, the server receives the linkage of the user's profile with the App's CID and SID and generates another trigger to send a message about the installation completion status to the users. The installation completion status can be sent via SMS text or via email, depending on the user profile set up by the user or system administrator.

Figure 3. (a) Shows the registration process to receive OTP after Bio-Guard is installed on the user's mobile device. (b) Shows the OTP confirmation process to receive SID after verification of the user's profile from the one available in the MHS database (DB).

Thus, the user's device gets the company's SID linked to the CID of the app. The combination of CID and SID along with user profile and IMEI creates a complex and unique identifier for user verification. In regular physical biometric authentication, the secret key is stored in the fingerprint sensor and is compared with the secret key stored in the security policy [48]. In the case of authentication via the Bio-Guard app, the combination of SID, CID, and the secret key stored in the fingerprint sensor is compared with the server-side secret key stored in the security policy, SID and CID. For existing users, the user profile of the web application and that of the app get in synch. For new users, their profile needs to be first created and verified by the system administrator team before users will receive OTP.

3.3. The Authentication Process of the Bio-Guard App

The web-based application to be accessed via the app had Oracle11g as a database. Therefore, the secure hashing algorithm 1 (SHA-1) was used for biometric authentication because certificates signed via the message digest 5 (MD5) algorithm face difficulty in HHTP/web environments [49]. The application was developed using a two-tier architecture model. Oracle11g was used as a back-end database that supports web-based biometric authentication via oracle adaptive access manager (OAAM) [50]. For experiments mentioned in this paper, biometric characteristics from specific users were previously captured, and the model templates were stored in the database.

The fingerprint sensor of the user's mobile devices was used only for training and getting aligned with the existing template models. Minutiae-based fingerprint features were used during the test for user authentication because those templates were available in the application database of the web application used for testing to authenticate the testers [51]. For a better understanding of the authentication logic of the application, a summary of the variables and parameters to be verified are declared below. We also describe here how the variables will be compared and validated for authentication to grant access to the back-end web application.

The user ID is associated with First + Last Name, DOB, Email, and Mobile device of the user, in the background web-application, i.e., FBAS. UID verification in the background web-application is a separate process of that application. For example, if a company decides to use employee ID or patient ID or company email as UID, then that ID must first exist before a user can register and receive OTP.

When Bio-Guard app is not installed on the server, it will not have authentication decision point 1 to check linkage of the CID and SID, as shown in Figure 4b, and the system will authenticate based on following parameters and grant web-access when the secret key stored in the security policy matches any of the following parameters:

- UID + P + O or
- UID + B

Figure 4. (a) Shows the process flow of SID confirmation and registration completion. (b) Shows the authentication process flow steps after registration completion in the Bio-Guard app.

This situation does not have an extra layer of authentication and also is not associated with SCI(ID); therefore, it is less secure and more prone to security issues.

During the installation process of the Bio-Guard app, as shown in Figure 2a–c, the system would require the CID and SID to get associated with the security policy. Therefore, it would authenticate based on the following parameters and grant web-access when the secret key stored in the security policy matches any of the following parameters:

- UID + B+ SCI(ID) or
- UID + P + O + SCI(ID)

During the registration process, as shown in Figure 3a,b, CID gets associated with SCI(ID); therefore, it would require OTP to confirm UID of the user. Once the Bio-Guard app is fully registered, and installation is completed, then a user needs only the following info to log in to the FBAS because the system has already authenticated the SCI(ID).

- UID +P +O or
- UID + B

The sequential flow, as shown in Figure 4b, has an extra layer of authentication along with SCI(ID); therefore, it is a more secure authentication process. As shown in Figure 4b, if a mobile device is not registered, only the biometric feature match will not grant access to the FBAS. Users will also need to confirm their identity via an additional step of OTP confirmation along with a fingerprint profile match. Therefore, Bio-Guard as MoLaBSS would be a more secure framework.

4. Experimentation

Though the sources of research and examples cited in this paper are from many countries, the experiment and implementation aspect of the app is primarily scoped from the perspective of health care service providers of those countries where necessary infrastructure, rules, and procedures are in place to start with. The experimentation section has been subdivided into seven subsections: (4.1) the methodology of test population sampling; (4.2) testing environment; (4.3) performance test of the app; (4.4) design, functionality and usefulness test of the app; (4.5) overall usability evaluation of the app on the SUS scale; (4.6) test for false rejection rate (FRR), false acceptance rate (FAR), and equal error rate (EER); (4.7) national level survey to test the success of the MoLaBSS in a realistic hospital environment.

4.1. The Methodology of Test Population Sampling

Testing with biometric data is a very sensitive issue. It was challenging to find and convince test users and obtain their profiles in a web-based application. We could engage 110 employees of the project sponsoring organizations whose fingerprint information was already available in the database of the Fingerprint-Based Attendance System (FBAS) for another project. From cost, convenience, and time-effectiveness perspectives, the expert/judgment sampling method was used so that the expertise of selected experienced subjects/experts could be used to represent the response of a larger unknown population, which was beyond the available resource pool of this project. Judgment sampling is not always preferred but is technically and conceptually the most recommended approach to select useful samples who know about the process, performance, and the impact of changes over a longer period [52]. It was assumed that each expert tester/user would represent a response of 10–15 non-expert users. Thus, the quality of the response from a population of 110 experienced testers with experience in the usage of biometric sensors can be equivalent to about 1100 people without such experience. Moreover, initially, 250 people were interviewed for experimentation, out of which 200 were selected, but only 110 testers agreed to allow using their biometrics profile in the web application and signed the “privacy consent form” (PCF). The testers taken for the experiment were from Mumbai, India—a city that is very cosmopolitan, with people from diverse socio-economic, cultural, and linguistic

backgrounds. These testers had either a graduate or master's degree in engineering or science. Some of those testers also had software development and testing work experience in western countries.

The survey was conducted at the national level in the patient registration halls of the top-notch hospital environment in five major metro cities of India—i.e., Mumbai, Delhi, Chennai, Bangalore, and Kolkata. These hospitals are known to have registrants from many countries who come here as medical tourists for high-quality, low-cost treatments. The original sample population chosen for screening and interview was 600, of which 450 had experience in the usage of fingerprints and or biometric devices. Out of 450, only 300 agreed to spend time learning the process, sign the “FAP” and take part in the survey. Most of these users had either a graduate or master's degree in science. Thus, in this high-quality sample population, 300 users who were screened and interviewed may be considered to represent about 3000 members of the general population. Remaining demographics, i.e., mean age and the number of males, females, etc., of the sample population are described in the respective experimental situation.

4.2. Testing Environment

This subsection provides some brief background information about the Bio-Guard app and the back-end web application, which was being accessed by the testers via the app. Like many other applications, this FBAS had three checkboxes in users' profiles to send them alerts/information for verification. One checkbox was for agreeing for the MFA to send a one-time password (OTP) for user verification after successful authentication of the password. One checkbox was for sending a short text alert via SMS on the user's registered mobile number. The third checkbox was for sending a detailed alert message via email on the user's registered email address. The sample of SMS and an email notification message is given in Appendix A, which can be changed by the system administrator. Users were asked to choose all the checkboxes and agree to online agreement of the company about online data privacy if they wanted those services to get activated on their account. The company also provided an option to opt out of those services if users wanted to. This paper assumes that the readers already know how an identity verification process works on a standard fingerprint web application, so this verification process is not being detailed here.

Before the installation of the Bio-Guard app, users could log in to the FBAS online remotely using their regular password. Anyone could log in to the user account with stolen UIDs and passwords. Users could log in to the FBAS with their fingerprint but were not using it for fear of spoofing and security. Anyone with a spoofed fingerprint could log in to the user's profile and change the password and phone number to receive a notification.

This was possible long before the user could get to know about it. However, once the Bio-Guard app is connected to the FBAS, the hacker needs a registered device. A user would not be allowed to use any unregistered Fingerprint-Sensor-Enabled-Mobile (FSEM) device. Users can install the client side of the Bio-Guard app on multiple FSEMs, but the number of FSEM devices allowed would be determined in the company's server database by the system administrator.

Moreover, the user would get a notification about another FSEM device along with the timestamp, location, and device ID of the new device during the app installation process itself. It would help the users to take immediate preventive action by reporting it to the system administrator to disable FSEM access and change the password if there is any doubt about it being compromised or lock the account till further verification. It would make the hackers' job more difficult, and users' accounts more secure.

If anyone else tries to use a spoofed fingerprint of the user from a stolen but registered FSEM device, then users would get a notification about another FSEM being used, provided the notification profile is not changed yet. This would help trace the stolen device location, and the user can disable the remote login option in the FBAS and contact the system admin to deregister the stolen FSEM device from the server database. This would generate confidence in users to use their mobile devices for biometric authentication.

The app requires CID linkage to the SID to complete the installation process. Therefore, the complete installation has three steps. (a) Download and install the setup. This creates an icon Bio-Guard on the mobile device's screen to a single tap and expands and registers the app. (b) Fill up relevant data in the fields of the Bio-Guard screen, as shown in Figure 2c, to register the App's CID linked to IMEI. It will connect the user's profile to the App's CID and will send the SID to the registered users, on confirmation of OTP. (c) Users single-tap the submit button after entering received SID, and, on the verification of the SID, users get a notification about the completion of the installation process. After installation and registration, as shown in Figure 2b, Figure 3a,b and Figure 4a, the registered users will get information from the server, provided a notification option is saved in the user's profile. The server side of the Bio-Guard app can be configured to send notifications about the first installation on a device so that users can get a minimal notification if they choose. Once an installation completion status message is received, users can securely log in to the web application connected to Bio-Guard either via Biometric ID in one step or via mobile password plus the OTP received from the application in two steps.

4.3. Performance Test of the App

Phan et al. [53] proposed a transport-protocol-based performance evaluation for mobile apps. Oliveira et al. [54] proposed an energy-consumption-based method of evaluation for mobile apps. Many users prefer security over performance, and many vice-versa. For this reason, we tested the response time and battery consumption performance of the Bio-Guard app.

To achieve the diversity objective for this experiment, we tested the performance on three smartphones of different capacity, in terms of RAM (random access memory), battery power, CPU (central processing unit), and hard disk drive (HDD) [55]. We deliberately chose one legacy, one latest and one somewhat older mobile phone to add variety in the year of manufacturing. We chose a legacy phone with a lower capacity to test the backward compatibility of the app. Still, we did not go beyond 2017 because phones older than that may not have very reliable fingerprint sensors. Table 1 shows the relevant specifications.

Table 1. Mobile phones used in the performance test of the app and their main relevant features.

Mobile Phone	Year Made	HDD Capacity	RAM in GB	CPU in GHz	Battery in (mAh)
MP1	2019	32	16	2.3 Octa-core Quad	3000
MP2	2018	16	8	1.4 Quad-Core	2000
MP3	2017	8	4	1.2 Quad-Core	2000

To test the performance, we deactivated all other apps and ran the Bio-Guard app on a fully charged battery for 30 min; allowing maximum possible time, the app may be required to be in use at a time. Mobile phone MP1 consumed 151 milliampere hours (mAh) of energy in 30 min, which equals to about 3.5% of the full battery power. Mobile phone MP2 consumed about 11% while MP3 consumed about 6%, as shown in Figure 5.

We measured response time for three important functions of the app: i.e., login, minor address update, and closing of the app while WhatsApp was active in the background. The result is shown in Figure 6. The app took a maximum response time of 1600 milliseconds (ms) for login by the legacy mobile phone (MP3) and a minimum response time of 800 ms for minor address update by the latest mobile phone (MP1). As per the experiment, the app's response time seems reasonable, though it may require some improvement based on broader test scenarios.

Figure 5. This figure displays the comparative battery energy consumption of the Bio-Guard app for 30 min, on the mobile phones MP1, MP2, and MP3. The X-axis represents duration, in minutes, and the Y-axis represents battery energy consumption in milliamper hours (mAh).

Figure 6. This figure displays the comparative response time of the Bio-Guard app for three main functions, on mobile phones MP1, MP2, and MP3. The X-axis represents the three main functions of login (1), minor address update (2), and closing the app (3). The Y-axis represents response time in milliseconds (ms).

4.4. Design, Functionality and Usefulness Test of the App

Testing with live biometric data of a large number of people is a very sensitive issue. It was challenging to find and convince many test users. However, we tried our best to make experimental methods as representative and scientific as feasible in our given circumstances. We could engage 110 employees of the project sponsoring organization, whose fingerprint information was already available in the database of FBAS, for another project.

The participants had fingerprint access to one of the following Android phones, e.g., Samsung Galaxy Note-9, Samsung Galaxy S9, Google Pixel 2 XL, OnePlus 6, Vivo X21 UD, Huawei P20 Pro, Huawei Honor 8 Pro, Sony XperiaXz Premium, Samsung Galaxy S8 Plus, Samsung Galaxy Note 8, Samsung Galaxy S7 Edge, One Plus 5T, and Asus Zen Fone Ar.

The median age of participants was 22.3 years ($SD = 4.28$). There were four male (40%) and six female (60%) participants. These test users had previous experience of fingerprint-based web application login.

To test the comparative functionalities of the app, we prepared ten experimental scenarios on the model of the SUS usability scale [56]. We made one question for each experimental situation and asked

users to mark their rating on a five-point Likert scale. Odd-numbered experimental items 1, 3, 5, 7, and 9 were direct questions and were scored as scale position minus 1. Even-numbered items, i.e., 2, 4, 6, 8, and 10 were inverted questions and were scored as 5 minus the scale position. To make the scoring easy, we converted the 10 “experimental questions” (Expt Q1 to Expt Q10) and presented them in the following statement format, as shown in Table 2. The total score of each user was multiplied by 2.5 to convert the overall SUS score to the range of 0 to 100.

Table 2. Ten experimental questions (Expt Q) prepared on the format of the SUS scale but presented in statement format for easy scoring.

Expt Q1. It was easy to install, register, and complete the setup of the app.
Expt Q2. It was difficult to register to the “Bio-Guard” app via Google account.
Expt Q3. Getting the server-side installation ID (SID ¹) was quick, on verification of the OTP ² .
Expt Q4. It was difficult to load the app on the screen after a single tap of the “Bio-Guard” icon.
Expt Q5. The visual clarity and alignment of the app on the screen was good and convenient to use after it loaded on a single tap of the “Bio-Guard” icon.
Expt Q6. It was difficult to remotely log in to the FBAS ³ application and access the personal profile via the fingerprint sensor of the mobile, after installing the “Bio-Guard” app.
Expt Q7. I felt more confident in the “Bio-Guard” app when I could not log in to the FBAS, using the fingerprint sensor of an unregistered mobile.
Expt Q8. I lost confidence in using fingerprint sensors to remotely access my record after I received a notification when I logged in to the FBAS from another mobile device via OTP ² .
Expt Q9. All the box fields and buttons of the “Bio-Guard” app functioned with ease and as expected.
Expt Q10. It was difficult to close the app with a single tap of the “home/close” button of the mobile device.

¹ SID—server installation ID, ² OTP—one-time password, ³ FBAS—fingerprint-based attendance system.

The test installation set up of the app is loaded to GitHub and can also be requested by writing an email to the copyright holder mentioned in GitHub [57]. The results were analyzed and interpreted on the model of the SUS score interpretation guide. A SUS score above 68 is considered above average. For example, a raw SUS score of 71 in this experiment converts to a percentile rank of 67% [58].

Before the experimentation, the server side of the application was already installed, and the installation ID was generated, e.g., sid1234. We asked users to log in to the FBAS via a physical local fingerprint scanner and select both the checkboxes mentioned above to get an SMS as well as an email alert and save their profile. We gave a printed copy of the sheets of the experimental questions to the group of 110 users, explained to them the scoring pattern, and briefed them on how the app would work. We gave the users a link to download the client side of the app.

4.4.1. Experiment 1

Was it easy to install, register, and complete the installation process of the Bio-Guard app?

Many users may not prefer to use an app that is difficult to set up. To test the speed and ease of installation of the app, we asked users to connect to the local wi-fi network and download the complete installation setup of the client-side of the Bio-Guard app from a web link to get it completely installed on their smartphones.

To complete the installation process of the app, users performed the steps shown in Figure 2b, Figure 3a,b and Figure 4a. Users downloaded and installed the client side of the Bio-Guard app, which generated CID and linked it to the IMEI of their mobile device and also created an icon Bio-Guard on the user’s mobile. Users single-tapped the Bio-Guard icon, expanded the app’s screen, entered the required data, and registered the installation of the Bio-Guard app to receive the OTP, as shown in Figure 2c. Users then verified their registered identity by confirming the OTP they received. Later,

users received SID, entered the SID, tapped the submit button, and received a message about the installation completion status.

We had explained installation and registration steps to the users to set the expectation right. We requested the users to consider each step as one installation of a regular app while comparing and rating this process of the app. We asked users to rate the ease of installation, as compared to other apps they use, on a scale of 1 to 5 (1 means “I strongly agree that installation of the Bio-Guard app was easy” and 5 means “I strongly disagree”).

4.4.2. Experiment 2

Was it difficult to register the Bio-Guard app via Google account?

For convenience purposes, many users prefer to use their Google or Facebook profiles to log in to apps. To test the functionality of the app to register via google API, during the installation process, we requested the users to sign up via their google account to receive the OTP. We then asked them to compare the response time with the response time of other apps when they log in to other apps via their google account.

We asked users to rate the response time, on the scale of 1 to 5 (1 means “I strongly agree that the app was very slow to log in via google account and took longer than expected time” and 5 means “I strongly disagree”).

4.4.3. Experiment 3

Was getting the server-side installation ID (SID) quick, on verification of the OTP?

Quickly getting the SID is a critical criterion to complete the installation process. Therefore, we evaluated the response time performance of the server in sending server-side installation ID on confirmation of the OTP, separately.

After the installation process was completed, as per experiment 1, we asked testers to compare the response time in getting the SID after they verified the OTP they received with other such processes they perform for other apps. We asked to rate the response time on the scale of 1 to 5 (1 means “I strongly agree that the server’s response time in sending SID was reasonably good” and 5 means “I strongly disagree”).

4.4.4. Experiment 4

Was it difficult to load the app on the screen after a single tapping of the Bio-Guard icon?

If an installed app does not load quickly to perform its functions, many users would be hesitant to use such an app. To test the loading speed of the app, we asked users to single-tap the “Bio-Guard” icon to load the app, to get expanded for use on their mobile screen, as shown in Figure 2c.

We asked users to rate the loading time as compared to the average loading time of other apps they use, on the scale of 1 to 5 (1 means “I strongly agree that the app was struggling to load and took longer than expected time” and 5 means “I strongly disagree”).

4.4.5. Experiment 5

Was the visual clarity and alignment of the app on the screen good and convenient to use after it loaded on a single tap of the Bio-Guard icon?

For correct data entry, comfort, and aesthetics, this aspect of the test is crucial. To test the visual alignment and clarity of the app, in the ease of operations, we asked the users to check the visual clarity of the app after it expanded on the mobile screen on a single tap of the Bio-Guard icon, as shown in Figure 2c. We requested users to cross-check the field’s text, button text and edges of the expanded app on their mobile, and compare it with other test users. We asked testers to make sure that the screen was not truncated on the sides, clear enough, and well-aligned for easy use, on a mobile screen.

We asked users to rate visual clarity and alignment on the scale of 1 to 5 (1 means “I strongly agree that the app’s visual clarity and alignment were okay to make it usable” and 5 means “I strongly disagree”).

4.4.6. Experiment 6

Was it difficult to remotely log in to the FBAS application and access the personal profile via the fingerprint sensor of the mobile, after installing the Bio-Guard app?

The whole purpose of this app is to securely and easily log in to a web application with an extra layer of security. This experiment was to test this core functionality. To check the ease of accessing a web application remotely, after adding an extra layer of protection with an app like “Bio-Guard,” we asked testers to log in via the fingerprint sensors of their mobile. Then we asked them to rate the ease of login as compared to other fingerprint-based web app logins on a scale of 1 to 5 (1 means “I strongly agree that it was difficult to log in via the Bio-Guard app” and 5 means “I strongly disagree”).

4.4.7. Experiment 7

Did you feel more confident in the Bio-Guard app when you could not log in to the FBAS, using the fingerprint sensor of an unregistered mobile?

The purpose of this app is to provide an extra layer of security to the application it is connected to, thus not allowing remote access to the application without the Bio-Guard app. To perform this regression testing, we asked users to log in remotely via another mobile device where the Bio-Guard app was not installed. Testers clicked on the FBAS icon, which proposed to enter their user ID and password, but they could not log in. Testers could not log in to the online version of the FBAS from a test laptop, either. We then asked users to rate their confidence level in the security of the Bio-Guard app on the scale of 1 to 5 (1 means “I strongly agree that login via Bio-Guard is very secure” and 5 means “I strongly disagree”).

4.4.8. Experiment 8

Did you lose your confidence in using fingerprint sensors to remotely access your record after you received a notification when you logged in to the FBAS from another mobile device via OTP?

Getting a notification before a hacker can login to a user’s account is critical for an alert user. We asked all 110 users to log off from the current finger-based attendance system and requested them to create their fingerprint profile on two test mobiles. The Bio-Guard client app was already installed on these two test mobiles with fingerprint sensors. We then requested them to log in to their finger-based attendance system using the fingerprint sensor of the test mobile. We asked them to check their personal mobile for the alert they received. They all got notification messages, as mentioned in Appendix A, that “someone logged in to your account in the FBAS from another device.” Then, we asked the testers to rate their confidence level in the usage of fingerprint devices remotely, on a scale of 1 to 5 (1 means “I strongly agree I have lost confidence in using fingerprint sensors, to remotely access a web-application” and 5 means “I strongly disagree”).

For privacy and security reasons, we asked the testers to delete their fingerprint profiles from these two test mobiles after the test.

4.4.9. Experiment 9

Did all the box fields and buttons of the Bio-Guard app function with ease and as expected?

Everyone wants to use the app without any bugs, so getting the input of users about the functioning of the app was important. At the time of testing of this aspect of the app, users had experienced the operation of almost all buttons and boxes of the app.

Therefore, we asked users to rate the overall ease of operation of the app’s buttons and boxes on a scale of 1 to 5 (1 means “I strongly agree that the app’s fields and buttons were easy to operate” and 5 means “I strongly disagree”).

4.4.10. Experiment 10

Was it difficult to close the app with a single tap of the home/close button of the mobile device?

People use mobile devices for multiple applications. The Bio-Guard app should not hog the memory and computing power of the device forever, just for extra security. To test the closure of the app and without it interfering with other regular functions of the mobile, we asked the test users to single-tap the home button of their smartphone while the app was loaded, and its expanded screen was visible on their mobile's screen. We asked users to rate the closure time of the app as compared to the average closing time of other apps they use, on the scale of 1 to 5 (1 means "I strongly agree that app was struggling to close and took longer than expected" and 5 means "I strongly disagree").

The comparative score on the SUS scale questions and of the 10 experimental questions prepared on the format of the SUS scale is plotted in the box plot, as shown in Figure 7.

Figure 7. This box plot displays the comparative average score of the Bio-Guard app on the SUS Scale and the scale prepared on the pattern of the SUS scale, for 110 experiments. The X-axis represents 110 expert test users, and the Y-axis represents the comparative score.

4.5. Overall Usability Evaluation of the App on the SUS Scales

At the end of the ten experimental scenarios mentioned in Section 4.2, we gave the users an hour break. We then collected their responses to the actual SUS questions to get the overall usability of the app.

The average and standard deviations of the response scores on the original SUS scale and that of the test scale prepared for the Bio-Guard app on the pattern of the SUS scale are displayed in Table 3 and presented in Section 5 (analysis of the experimental outcomes and national level survey results).

Table 3. A comparative score of the standard SUS Scale and 10 experiments.

Average/Mean SUS rating score of 110 Users	71.2
Average/Mean Experimental rating score on 110 users on SUS modeled experiment, performed for the Bio-Guard app	73.95
Std Dev of SUS rating score of 110 Users	4.342
STD Dev Experimental rating score on 110 users on SUS modeled experiment, performed for the Bio-Guard app	5.941

A comparative rating score of the standard SUS scale and 10 experiments performed for the Bio-Guard app is presented and rated on the model of the SUS scale for 110 expert test users.

4.6. Test for False Rejection Rate (FRR), False Acceptance Rate (FAR), and Equal Error Rate (EER)

As per the guidelines of Robertas et al. [31] and Jorgensen and Yu [59], we tested the app for false rejection rate (FRR), false acceptance rate (FAR), and equal error rate (EER). A lower EER means a more accurate application system. To reduce training time, we conducted this test after we conducted usability tests.

While FAR and FRR show if the app correctly identified the subject, EER specifies the values at which FAR and FRR become almost equal. The metrics were calculated as follows:

$$FRR = FR/AA$$

$$FAR = FA/IA$$

Here, FR (false rejections) is the number of falsely rejected verification attempts of a valid subject, and AA (authorized attempts) is the total number of authorized attempts. FA (false acceptance) is the number of the falsely accepted claims of an impostor as a genuine user. IA (impostor attempts) is the number of impostor attempts. The test yielded mean FAR and FRR values of 0.4200 and 0.3600, respectively. The resultant EER is 0.0600, as shown in Figure 8.

Figure 8. This figure displays the ERR at the intersection of FAR and FRR. The X-axis represents the similarity threshold, and the Y-axis represents error. In this case, at the similarity threshold of 54, the resultant ERR is 0.06.

4.7. National-Level Survey to Test the Success of the MoLaBSS in a Realistic Hospital Environment

A national-level survey was conducted on a sample population of 300 users in the realistic patient registration hall setting of the project-sponsoring hospital and its partners, using recorded video clips of the tests mentioned in Sections 4.3–4.5 [60]. The median (MD) age of participants was 25.3 years, with a standard deviation (SD) of 3.48. There were 120 male (40%) and 180 female (60%) participants. Some of these users, including the 100 testers of experiments Sections 4.3–4.5, plus 200 additional potential users of the app, had previous experience of fingerprint-based web application login. For privacy reasons and to protect the identity of the testers, the video clip of the recorded test used for this survey is not being made public but may be received from the project sponsoring organization.

We explained the experimental details to the users and then showed the video recorded for tests Sections 4.3–4.5. They responded to the survey questions, as shown in Table 4, on a five-point Likert scale for the server-specific add-on biometric security model (MoLaBSS) to enhance the usage of biometrics. The cumulative columnar bar graph shown in Figure 9 displays the survey results.

Table 4. Survey questions to test the success of the MoLaBSS.

(1) I felt more confident to use web-based applications via biometric devices when I saw that the testers could not login to the FBAS using the fingerprint sensor of an unregistered mobile device.
(2) The concept of getting the server-side installation ID (SID) only on the verification of the OTP is very useful for prompt preventive action.
(3) I think with this kind of add-on extra layer of security; biometric technologies can be more convenient and reliable than password-based access systems.
(4) I think MoLaBSS can make access to personal information more secure.
(5) After watching the performance of the Bio-Guard app, I think MoLaBSS can be useful for many groups of people.

Figure 9. This cumulative columnar bar graph displays the response of survey results of 300 expert users for the success of the MoLaBSS. The X-Axis shows the survey questions (1 to 5). The Y-Axis displays the number of users who responded on the Likert, from strongly agree to strongly disagree.

5. Analysis of the Experimental Outcomes and National level Survey Results

Experimental scores of the above tests were recorded and normalized, and the results are presented below. Figure 5 shows that the Bio-Guard app consumed an average of about 7% of the battery power when it ran alone for 30 min. Therefore, from the power/energy consumption perspective, it seems reasonable. However, it may require some improvements after testing in more situations.

Figure 6 shows that the average response time of the Bio-Guard app is approximately 1200 milliseconds per transaction, while WhatsApp was running in the background. This response time seems okay for a new app; however, more tests may be required on a broader testbed for a decisive conclusion about this.

The mean FAR and FRR values of 0.4200 and 0.3600, respectively, resulted in an EER of 0.0600 (6%) at the similarity threshold of 54, as can be seen in Figure 8. At the higher threshold, the ERR could be lowered, but in that case, FRR could have gone up. To maintain a balance, the threshold was kept at 54%, and resulted in a slightly higher ERR. This EER is reasonably low but needs further balancing and reduction.

The SUS score, between 70 and 80, is better than average [61]. As per Table 3, the average SUS score of the Bio-Guard app is 71.2. The average rating score of 10 experiments performed on the pattern of the SUS scale for the Bio-Guard app has a value of 73.95. Both these results suggest that the app's functionalities and performance are in an acceptable range, though the overall usability of the app is rated higher. The results also point towards the possibility of improving the app's usability by enhancing some of its functionalities.

The comparative SUS score and the rating score of 10 experiments modeled on the SUS scale, as shown in the box plot graph in Figure 7, indicate that the fluctuation in the score was user specific. In other words, the users who scored higher on the SUS-modeled experiment also scored high on SUS usability. This also indicates that the usability rating is a dependent function of factors such as how techno-savvy the users are and how efficient their smartphones are. It implies that, with some practice and training, the scores can be enhanced.

The majority of users in the population sample scored around 70 on both scales, which is near the middle of the SUS scale. In other words, the users who had a high score on the SUS Scale also had a high score on the scale prepared for 10 experiments performed on the pattern of the SUS scale for the Bio-Guard app, as shown in the box plot in Figure 7. This indicates that the personality of users taking these tests and their socio-economic background may influence the acceptance of biometric technologies. For example, some users may be early adopters and high-risk-takers, while others may be low risk-takers.

The survey results for the success of MoLaBSS mentioned in Section 4.7, as shown in the cumulative columnar bar graph, Figure 9, help extrapolate the following information:

- (a) Question 1: 86% (258 out of 300) of the users agreed that they felt more confident to use web-based applications via biometric devices when they saw that the testers could not login to the FBAS using the fingerprint sensor of an unregistered mobile device (MD:4.5, SD: 1.04).
- (b) Question 2: 83% (249 out of 300) of the users that responded agreed or strongly agreed that the concept of getting the server-side installation ID (SID) on the verification of the OTP is very useful for prompt action (MD: 5, SD: 1.16).
- (c) Question 3: 71.33% (214 out of 300) of the users that responded agreed or strongly agreed that, with this kind of “add-on” extra layer of security, biometric technologies could be more convenient and reliable than password-based access systems (MD: 4, SD: 1.22).
- (d) Question 4: 86.66% (260 out of 300) of the users that responded agreed or strongly agreed that the MoLaBSS could make access to personal information more secure (MD: 4, SD: 0.99).
- (e) Question 5: 76.66% (230 out of 300) of the users that responded agreed or strongly agreed that the MoLaBSS could be useful for many groups of people (MD: 4, SD: 1.21).
- (f) Overall, the survey results for the success of MoLaBSS are very favorable, with an average of 80.66% of the testers responding with agree or strongly agree to the above five survey questions.

6. Discussion on Implementation Opportunities and Challenges

Like any other novel solution framework, the successful implementation of MoLaBSS can provide many opportunities and may face a few challenges, as discussed below. The framework has strong potential to provide opportunities for SSO (single sign-on) with an add-on extra layer of security, especially for the healthcare industry. An increased sense of security in the usage of web-applications can influence a deeper level of cognition to learn and to adapt to new challenges.

6.1. Authentication/Validation Challenges

With the discovery of the usage of new sensors, new applications, testing, and validation tools in the light of the 5G network and beyond, the app may face new challenges and may also find many opportunities to grow.

6.2. Training, Awareness and Adaptation Challenges

So far, we have had the opportunity to develop and test the framework with Android-based devices. Its development, deployment, and testing on a broader sample population for iOS and Windows may face new challenges. People, especially in developing countries, need to be aware of newly available tools to be trained on.

6.3. Security and Privacy Challenges

The sense of security and confidence in new tools is often perceptual. The purpose of this framework is to encourage higher usage of biometric technologies, but, like other new models, it may take some time to build trust and confidence in the capabilities of this framework.

7. Conclusions and Future Perspectives

Based on the information analysis and the review of relevant literature and reports, biometrics seems to be one of the most useful technologies for multifactor authentication to verify the user's identity for data security. With the fast growth of biometrics as BaaS in cyber-society, especially for usage with IoT devices and related security vulnerabilities, it has gone high on the radar of many companies [62].

The literature review also indicates that, because of the low usage rate of biometrics, many organizations, especially small and medium segment healthcare companies, are not finding it very cost-effective to invest in robust biometric systems. Since many of them lack dedicated fraud prevention and security teams, they are becoming vulnerable to cyber-hackers. Therefore, the usage rate of biometrics must be increased to achieve economy of scale.

The architecture, design, flow diagrams, logic, and functional and operational aspects of the framework seem logical as an add-on for an extra layer of security to biometric-based web-applications. The experimental results for battery performance, response time, and regression tests of accessing web apps seem to have generated confidence in the testers. A reasonably low equal error rate (EER) of 6% and a score of 71% on the usability scale indicate the authenticity of the app as a testing tool. The survey for the success of the server-specific add-on biometric security layer model (MoLaBSS), has a very favorable score of 80%.

The experimental outcomes of the biometric mobile app and subsequent survey for the success of the MoLaBSS show reasonable potential for this kind of add-on layer of biometric security in increasing the participation of users in the usage of biometrics. Higher usage rates may make deployment of biometrics more cost-effective for many organizations to decrease their information security risk. However, the MoLaBSS must overcome a few challenges related to training, adaptation, and socio-cultural data privacy norms.

Author Contributions: B.S. is the primary author of this research, who conceptualized, conceived, and designed the mobile app software and experiments. N.S. performed project administration, data analysis, investigation, validation, and provided suggestions to improve the paper. Both authors were involved in experiments, data curation, review, and approval of the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by a joint pilot project between Nishaan Infotech US LLC, Nishaan Infotech India Pvt. Ltd., and Lifeline Hospital, Mumbai, Maharashtra, India. Some of the information is covered under the confidentiality agreement, so, the permission of the above three organizations would be required with reference to project number LLHMH.I.proj.852019. However, this article is not funded by any organization.

Acknowledgments: The author is very thankful to the anonymous reviewers and colleagues who provided insightful thoughts, valuable suggestions, and encouragement. The author is also grateful to Jo Quintanilla, University of Sydney, Australia, a digital project manager by profession, and P. Singh, Bombay University, India, for organizing information and proofreading the papers. The author remains the only person accountable for any omissions or mistakes.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

1 The SMS short message alert of the Bio-Guard app:

A security alert from the company abcd123 employee id ending xxx55.

Dear Mr./Ms. Abc.Xyz

We are letting you know that your employee record was accessed using device id, location ABC on date dd.mm.yyyy at hh.mm.ss (time).

If you don't recognize this activity or want to review your account activity, please check your registered email for details or contact your system admin team.

2 Detailed email alert message of the Bio-Guard app:

A security alert from the company abcd123 employee id ending xxxxx55.

Dear Mr./Ms. Abc.Xyz

We are letting you know your employee record was accessed using device id, location ABC on date dd.mm.yyyy at hh.mm.ss (time).

If you don't recognize this activity or want to review your account activity, go to www.abcdefg.com or contact your system admin team.

Remember: We never ask for personal information such as employee id, card PIN, or social security number or Tax-ID, driving license number, etc. in emails. However, if you think an email is suspicious, don't click on any links. Instead, forward it to www.abcdefg.com and delete it.

Please note that you are receiving this email as per your selection choice and service agreements, whether or not you elect to receive such alerts. Read our privacy notice and don't reply directly to automatically generated messages. Abcd Ltd. © 2019, all rights reserved. This email was sent to abc.xyz.123@gmail.com.

References

1. Spirina, K. Biometric Authentication: The Future of IoT Security Solutions, 5 July 2018. 2020 IoTEVOLUTIONWORLD. Available online: <https://www.iotevolutionworld.com/iot/articles/438690-biometric-authentication-future-iot-security-solutions.htm> (accessed on 3 June 2020).
2. Abomhara, M.; Køien, G. Cybersecurity and the Internet of Things: Vulnerabilities, threats, intruders and attacks. *J. Cyber Secur.* **2015**, *4*, 65–88. [CrossRef]
3. Ramírez-López, F.; Varela-Vaca, Á.J.; Ropero, J.; Luque, J.; Carrasco, A. A Framework to Secure the Development and Auditing of SSL Pinning in Mobile Applications: The Case of Android Devices. *Entropy* **2019**, *21*, 1136. [CrossRef]
4. Jalali, M.; Kaiser, J.P.; Jarrett, M.; Ghaffarzadegan, N. Cybersecurity in Hospitals: A Systematic, Organizational Perspective. *J. Med. Internet Res.* **2018**, *20*, e10059. [CrossRef]
5. Cherqi, O.; Mezzour, G.; Ghogho, M.; El Koutbi, M. Analysis of Hacking Related Trade in the Darkweb. In Proceedings of the 2018 IEEE International Conference on Intelligence and Security Informatics (ISI), Miami, FL, USA, 9–11 November 2018; pp. 79–84.
6. Richard, G. Most Common and Hackable Passwords on the Internet, 12 September 2013. Available online: <https://www.telegraph.co.uk/technology/internet-security/10303159/Most-common-and-hackable-passwords-on-the-internet.html> (accessed on 3 June 2020).
7. Yıldırım, M.; Mackie, I. Encouraging users to improve password security and memorability. *Int. J. Inf. Secur.* **2019**, *18*, 741–759. [CrossRef]
8. O'Connor, F. *What Happens If Your Biometrics Are Stolen?* Veridium Ltd.: London, UK, 2019; Available online: <https://www.veridiumid.com/blog/biometric-mythbusters-stolen-fingerprints-mean-identity-theft/> (accessed on 3 June 2020).
9. Joy, K. Biometrics in Healthcare: How It Keeps Patients and Data Safe? 23 December 2019. Available online: <https://healthtechmagazine.net/article/2019/12/biometrics-healthcare-how-it-keeps-patients-and-data-safe-perfcon> (accessed on 3 June 2020).
10. Varela-Vaca, Á.J.; Gasca, R.M.; Ceballos, R.; Gómez-López, M.T.; Torres, P. CyberSPL: A Framework for the Verification of Cybersecurity Policy Compliance of System Configurations Using Software Product Lines. *Appl. Sci.* **2019**, *9*, 5364. [CrossRef]
11. Arthur, C. iPhone 5S Fingerprint Sensor Hacked by Germany's Chaos Computer Club. The Gaurdian, 23 September 2013. Available online: <https://www.theguardian.com/technology/2013/sep/22/apple-iphone-fingerprint-scanner-hacked> (accessed on 3 June 2020).

12. Razaghpanah, A.; Sundaresan, S.; Niaki, A.A.; Amann, J.; Vallina-Rodriguez, N.; Gill, P. Studying TLS usage in Android apps. In Proceedings of the 13th International Conference on Emerging Technologies. (CoNEXT 2017), Ingeon, Korea, 12–15 December 2017; pp. 350–362.
13. Hacking Responsible for 83% of Breached Healthcare Records in January. HIPAA Journal, Last modified 1 March 2018. Available online: <https://www.hipaajournal.com/hacking-responsible-83-breached-healthcarerecords-january> (accessed on 3 June 2020).
14. Srinivasan, A.; Nguyen, A.; Tarlecki, R. STUMP–STalling offline password attacks Using pre-hash ManiPulations. In Proceedings of the 2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS), Melbourne, Australia, 14–17 December 2015; pp. 306–313.
15. McDonough, B.R. *Cyber Smart: Five Habits to Protect Your Family, Money, and Identity from Cyber Criminals*; John Wiley & Sons: Hoboken, NJ, USA, 2019.
16. Kapko, M.; Finnegan, M. What is Windows Hello? Microsoft’s Biometrics Security System Explained. *Computerworld*. 26 November 2018. Available online: <https://www.computerworld.com/article/3244347/what-is-windows-hello-microsofts-biometrics-security-system-explained.html> (accessed on 3 June 2020).
17. Avinash, S.; Wu, J.; Shi, J. Secure android covert channel with robust survivability to service provider restrictions. *Int. J. Secur. Netw.* **2017**, *12*, 27–39.
18. Gavrilova, M.L.; Monwar, M. *Multimodal Biometrics and Intelligent Image Processing for Security Systems*; IGI Global: Hershey, PA, USA, 2013; pp. 48–148. [CrossRef]
19. Cheng, J. Securing IP Surveillance Cameras in the IoT Ecosystem. *Trend Micro IoT Security*. 18 April 2018. Available online: <https://www.trendmicro.com/vinfo/mx/security/news/internet-of-things/securing-ip-surveillance-cameras-in-the-iot-ecosystem> (accessed on 3 June 2020).
20. Aghili, S.F.; Mala, H.; Shojafar, M.; Peris-Lopez, P. LACO: Lightweight Three-Factor Authentication, Access Control and Ownership Transfer Scheme for E-Health Systems in IoT. *Futur. Gener. Comput. Syst.* **2019**, *96*, 410–424. [CrossRef]
21. Meng, W.; Wong, D.S.; Furnell, S.; Zhou, J. Surveying the Development of Biometric User Authentication on Mobile Phones. *IEEE Commun. Surv. Tutor.* **2014**, *17*, 1268–1293. [CrossRef]
22. Chen, Y.; Sun, J.; Zhang, R.; Zhang, Y.; Yimin, C. Your song your way: Rhythm-based two-factor authentication for multi-touch mobile devices. In Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM), Hong Kong, China, 26 April–1 May 2015; pp. 2686–2694.
23. Ferrag, M.A.; Maglaras, L.; Derhab, A.; Janicke, H. Authentication schemes for smart mobile devices: Threat models, countermeasures, and open research issues. *Telecommun. Syst.* **2019**, *73*, 317–348. [CrossRef]
24. He, D.; Bu, J.; Chan, S.; Chen, C.; Yin, M. Privacy-Preserving Universal Authentication Protocol for Wireless Communications. *IEEE Trans. Wirel. Commun.* **2010**, *10*, 431–436. [CrossRef]
25. Li, W.; Gu, Q.; Zhao, Y.; Wang, P. Breaking Two Remote User Authentication Systems for Mobile Devices. In Proceedings of the 2017 IEEE 3rd International Conference on Big Data Security on Cloud (Bigdatasecurity), Ieee International Conference on High Performance and Smart Computing (HPSC), and Ieee International Conference on Intelligent Data and Security (IDS), Beijing, China, 26–28 May 2017; pp. 37–42.
26. Traore, I.; Woungang, I.; Nakkabi, Y.; Obaidat, M.S.; Ahmed, A.A.E.; Khalilian, B. Dynamic Sample Size Detection in Learning Command Line Sequence for Continuous Authentication. *IEEE Trans. Syst. Man Cybern.* **2012**, *42*, 1343–1356. [CrossRef] [PubMed]
27. Mondal, S.; Bours, P. Continuous Authentication in a real world settings. In Proceedings of the 2015 Eighth International Conference on Advances in Pattern Recognition (ICAPR), Kolkata, India, 4–7 January 2015; pp. 1–6.
28. Buduru, A.B.; Yau, S.S. An Effective Approach to Continuous User Authentication for Touch Screen Smart Devices. In Proceedings of the 2015 IEEE International Conference on Software Quality, Reliability and Security, Vancouver, BC, Canada, 3–5 August 2015; pp. 219–226.
29. Mondal, S.; Bours, P. Continuous authentication and identification for mobile devices: Combining security and forensics. In Proceedings of the 2015 IEEE International Workshop on Information Forensics and Security (WIFS), Rome, Italy, 16–19 November 2015; pp. 1–6. [CrossRef]
30. Aghili, S.F.; Mala, H.; Peris-Lopez, P. Securing Heterogeneous Wireless Sensor Networks: Breaking and Fixing a Three-Factor Authentication Protocol. *Sensors* **2018**, *18*, 3663. [CrossRef] [PubMed]
31. Damaševičius, R.; Maskeliūnas, R.; Kazanavicius, E.; Woźniak, M. Combining Cryptography with EEG Biometrics. *Comput. Intell. Neurosci.* **2018**, *2018*, 1867548. [CrossRef] [PubMed]

32. Wu, G.; Wang, J.; Zhang, Y.; Jiang, S. A Continuous Identity Authentication Scheme Based on Physiological and Behavioral Characteristics. *Sensors* **2018**, *18*, 179. [CrossRef] [PubMed]
33. Bamasag, O.O.; Youcef-Toumi, K. Towards continuous authentication in the Internet of Things based on secret Sharing Scheme. In Proceedings of the WESS'15: Workshop on Embedded Systems Security, Amsterdam, The Netherlands, 4–9 October 2015; pp. 1–8.
34. Vhaduri, S.; Poellabauer, C. Multi-Modal Biometric-Based Implicit Authentication of Wearable Device Users. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 3116–3125. [CrossRef]
35. Li, S.Z.; Jain, A. *Encyclopedia of Biometrics*; Chapter Passive Biometrics; Springer: Boston, MA, USA, 2009; pp. 46–65. [CrossRef]
36. Olanrewaju, L.; Oyebiyi, O.; Misra, S.; Maskeliūnas, R.; Damaševičius, R. Secure Ear Biometrics Using Circular Kernel Principal Component Analysis, Chebyshev Transform Hashing and Bose–Chaudhuri–Hocquenghem Error-Correcting Codes, January 2020. Available online: <https://link.springer.com/article/10.1007%2Fs11760-019-01609-y> (accessed on 3 June 2020).
37. Damaševičius, R.; Maskeliūnas, R.; Venckauskas, A.; Woźniak, M. Smartphone User Identity Verification Using Gait Characteristics. *Symmetry* **2016**, *8*, 100. [CrossRef]
38. Spanakis, E.G.; Spanakis, M.; Karantanas, A.; Marias, K. Secure access to patient's health records using SpeechXrays, a multi-channel biometrics platform for user authentication. In Proceedings of the annual international conference of the IEEE Engineering in Medicine and Biology Society, Orlando, FL, USA, 16–20 August 2016; pp. 2541–2544.
39. Spanakis, M.; Manikis, G.; Porwal, S.; Spanakis, E.G. Developing a context-dependent tuning framework of multi-channel biometrics that combine audio-visual characteristics for secure access in the eHealth platform for osteoarthritis management. In Proceedings of the 7th EAI International Conference on Wireless Mobile Communication and Healthcare, Vienna, Austria, 14–15 November 2017.
40. McCool, C.; Marcel, S.; Hadid, A.; Pietikäinen, M.; Matějka, P.; Cernock, J.; Poh, N.; Kittler, J.; Larcher, A.; Lévy, C.; et al. Bi-Modal Person Recognition on a Mobile Phone: Using Mobile Phone Data. In Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops, Melbourne, Australia, 9–13 July 2012; pp. 635–640.
41. Manikis, G.C.; Spanakis, M.; Spanakis, E.G. Personalized Mobile eHealth Services for Secure User Access Through a Multi Feature Biometric Framework. *Int. J. Reliab. Qual. E-Healthc.* **2019**, *8*, 40–51. [CrossRef]
42. Syed, Z.; Helmick, J.; Banerjee, S.; Cukic, B. Touch gesture-based authentication on mobile devices: The effects of user posture, device size, configuration, and inter-session variability. *J. Syst. Softw.* **2019**, *149*, 158–173. [CrossRef]
43. Ferrag, M.A.; Maglaras, L.; Janicke, H.; Jiang, J.; Shu, L. Authentication Protocols for Internet of Things: A Comprehensive Survey. *Secur. Commun. Netw.* **2017**, *2017*, 6563953. [CrossRef]
44. Jain, A.K.; Ross, A. Bridging the gap: From biometrics to forensics. *Philos. Trans. R. Soc. B Biol. Sci.* **2015**, *370*, 20140254. [CrossRef]
45. Fernández-Cerero, D.; Varela-Vaca, Á.J.; Fernández-Montes, A.; Gómez-López, M.T.; Álvarez-Bermejo, J.A. Measuring data-centre workflows complexity through process mining: The Google cluster case. *J. Supercomput.* **2019**, *76*, 2449–2478. [CrossRef]
46. Juned, M. How to Get Current GPS Coordinates Location Android Programmatically. Available online: <https://www.android-examples.com/get-current-gps-coordinates-location-android-programmatically> (accessed on 3 June 2020).
47. Juned, M. How to Get Facebook Login User Data ID, First Name, Last Name, Email, Gender, Link, Locale and Account Verified Status Programmatically, May 2017. Available online: <https://www.android-examples.com/facebook-login-graph-api-get-user-info> (accessed on 3 June 2020).
48. Configuring Identix Biometric Authentication. Available online: https://docs.oracle.com/cd/F49540_01/DOC/network.815/a67766/07_ident.htm (accessed on 3 March 2019).
49. Using the Fingerprint Certificate Mapper. Available online: <https://docs.oracle.com/cd/E19476-01/821-0506/using-fingerprint-cert-mapper.html> (accessed on 5 June 2019).
50. Device Fingerprinting and Identification, Oracle Adaptive Access Manager (OAAM). Available online: https://docs.oracle.com/cd/E40329_01/admin.1112/e60557/finger.htm#AAMAD6186 (accessed on 5 June 2019).

51. Roy, T.K.; Roy, T.K. Fingerprint Acquisition & Verification on Mobile Devices. In Proceedings of the 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2), Rajshahi, Bangladesh, 8–9 February 2018; pp. 1–5.
52. Perla, R.J.; Provost, L.P. Judgment sampling: A health care improvement perspective. *Qual. Manag. Healthc.* **2012**, *21*, 169–175. [[CrossRef](#)] [[PubMed](#)]
53. Phan, K.A.; Tari, Z.; Bertok, P. A benchmark on soap’s transport protocols performance for mobile applications. In Proceedings of the 2006 ACM Symposium on Applied Computing–SAC’06, Dijon, France, 23–27 April 2006. [[CrossRef](#)]
54. Oliveira, W.; Oliveira, R.; Castor, F. A Study on the Energy Consumption of Android App Development Approaches. In Proceedings of the 2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR), Buenos Aires, Argentina, 20–21 May 2017; pp. 42–52. [[CrossRef](#)]
55. Afzaal, M.; Usman, M.; Fong, A.C. Tourism Mobile App with Aspect-Based Sentiment Classification Framework for Tourist Reviews. *IEEE Trans. Consum. Electron.* **2019**, *65*, 233–242. [[CrossRef](#)]
56. Brooke, J. *SUS: A Quick and Dirty Usability Scale*; Jordan, P.W., Thomas, B., Weerdmeester, B.A., McClelland, I.L., Eds.; Usability Evaluation in Industry; Taylor & Francis: London, UK, 1996.
57. Nishaan Bio-Guard_Biometrics_App, 2019. Copyright: GitHub, Inc., 2019. Available online: https://github.com/NishaanGHac/Bio-Guard_Biometrics_app (accessed on 3 June 2020).
58. Jeff Sauro, M.E. Assuring Usability with the System the System Usability Scale (SUS), 2 February 2011. Available online: <https://measuringu.com/sus/> (accessed on 3 June 2020).
59. Jorgensen, Z.; Yu, T. On mouse dynamics as a behavioral biometric for authentication. In *Proceedings of the 6th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications–DIVANet’17*; Association for Computing Machinery (ACM): New York, NY, USA, 2011; pp. 476–482.
60. Lee, D.; Park, J.; Song, M. An App-Based Authoring System for Personalized Sensory Stimulation of Children With Developmental Disabilities. *IEEE Access* **2017**, *5*, 10583–10593. [[CrossRef](#)]
61. Bangor, A.; Kortum, P.; Miller, J. Determining what individual SUS scores mean: Adding an adjective rating scale. *J. Usability Study* **2009**, *4*, 114–123.
62. Conn, S. Gartner Identifies the Top 10 Internet of Things Technologies for 2017 and 2018, 23 February 2016. Available online: <https://www.gartner.com/en/newsroom/press-releases/2016-02-23-gartner-identifies-the-top-10-internet-of-things-technologies-for-2017-and-2018> (accessed on 3 June 2020).

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).