International Journal of Engineering &Technology, 8 (3) (2019) 193-204

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



Fingerprint and location based multifactor authentication for mobile applications

Norah Abdullah Aldumiji [1] *, Esam Ali Khan [1]

[1] Umm Al-Qura University, Kingdom of Saudi Arabia, Makkah
**Corresponding author E-mail: n.a.u.d@hotmail.com*

Abstract

Authentication, which involves the verification of identity, is one of the most important security features. It usually depends on three factors: something you know (knowledge), something you have (token) and something you are (biometrics). In this paper, we propose the use of biometrics (fingerprints) with a fourth factor, namely location (i.e., where you are), in order to develop a privacy- friendly multi-factor authentication scheme suitable for smartphone applications.

Keywords: *Authentication; Biometrics; location; Multifactor; Smartphone.*

1. Introduction

Phone applications are now becoming essential parts of our everyday lives. They are now integral aspects of our day-to-day activities starting from being woken by digital alarms, communicating through video telecommunications, socialising via social networks, purchasing groceries from an online store, and managing our finances using online banking services. Furthermore, an increasing number of activities are now becoming manageable through phone applications.

However, with the increased use of these applications that hold sensitive data and perform critical functions, security has become a necessity. Consequently, one of the essential aspects of security is authentication, which is "proof of identity, a process by which an entity provides acceptable proof that the identity of which the entity claims is in fact an identity that the entity is entitled to use" [1].

Authentication is usually performed based on three types of factors. The first is using something you know, namely knowledge, like a password. the second is something you have, namely a token that can either be designed explicitly for authentication like smart cards or tokens that have other primary functionality but can also be used for authentication, such as phones. the third is something you are, namely your behavioural or physical traits, such as a fingerprint.

although these factors have been sufficient for most cases, other factors could also be used, such as WHERE YOU ARE. An example of this is Location-Based Authentication, which can be useful information for authentication. For example, a hacker who lives in Russia should not be able to log in to the account of a user who lives in Saudi Arabia.

User location can provide many benefits, one of which is ensuring that critical operations can only be performed at approved physical locations, such as the remote control of critical systems. Similarly, sensitive information can only be accessed at these locations, like a company's financial transaction history. Another benefit is the ability to control the access to a physical site. Furthermore, user location can also be used to prevent spoofing, which is a major threat to network security and can prevent connection hijacking by continuously tracking user location. "Location-based authentication has the effect of grounding cyberspace in the physical world so that the physical locations of network entities can be reliably determined" [2].

Additionally, it can be used for security audit services "where security audit information is enhanced with the location of an entity or entities that are involved in an audited action" [3]. This can be particularly useful in cybercrime investigations by providing evidence that can be used for convicting or proving innocence, or for locating offenders. Furthermore, many potential intruders will be deterred if they know their location will be audited and their anonymity will be breached.

User location cannot be used for authentication on its own, but combining it with any of the other authentication factors can boost security and create Multi-Factor Authentication (MFA). MFA "is a secure process of authentication which requires more than one authentication technique chosen from independent categories of credentials" [4]. The more factors used, the better the security will be, bearing in mind that it is important to establish a good balance between security and usability.

Since smartphones include a wide range of technologies such as fingerprint readers and location-sensing technologies like GPS, we propose that these technologies can be used to implement MFA that will provide more security for smartphone applications while preserving their usability.

Nevertheless, both the fingerprint and location of the user are sensitive information, and using user fingerprint and tracking location can lead to privacy risks. An unauthorised party should not be able to access such information. Thus, this scheme must also maintain the privacy of the user.

Copyright © 2019 Norah Abdullah Aldumiji, Esam Ali Khan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

2. Related work

We will examine privacy friendly fingerprint authentication, location based authentication and MFA that have either fingerprint, location factor or both.

2.1. Fingerprint authentication

A case study for cancelling fingerprints using one-way transformations is presented in in [5] and [6]. The authors implemented the transformations in several ways including Cartesian, Radial, Polar and Functional transformation, and then they compared their relative merits empirically.
Hashing the fingerprint minutia using symmetric hash functions is proposed in [7]. Their approach has two assumptions. First, the location of the singular points is usually unstable and it is therefore undependable. Second, there is no pre-alignment between the test and the stored fingerprint templates. Their work provides secure fingerprint representation and matching. They extended the work in [8] by using a combination of symmetric hash functions instead of only one. This resulted in an increase in the security of fingerprint matching by an exponential factor.
A protocol that reports all users whose distance to the submitted finger code is under a given threshold is proposed in [9]. This work differs from previous works related to privacy preserving fingerprint as it usually returns the best matching user in the database.
A privacy-preserving fingerprint authentication based on minutiae representation is presented in [10] in which the advantages of Yao's classic Garbled Circuit (GC) protocol were used to develop a technique that is accurate, efficient and scalable.

2.2. Location based authentication

One of the first studies that highlighted the important use of location-based authentication in improving security is [11], in which the authors demonstrated how user location can be used to control the access to sensitive systems or information and how it can be used to prevent network breaches or to deter intruders. In their methods, they proposed the use of what they called a location signature, which works in the following manner. If a user wants access to a protected resource, his/her location should be provided to the server. To verify the user, the server compares the received location with its location and then grants the access. Nevertheless, in their approach, they proposed the use of a technology called cyberLocator, whereby users need to own a particular kind of GPS sensor that is not user-friendly.
Enhancing the knowledge factor by using the additional factors of possession and location is proposed in [1]. In this context, the location is obtained from mobile telephone network and then used to improve the access control and to provide an audit of the information.
An access control mechanism that is location-aware is proposed in [12]. However, it was specifically designed for controlling access to wireless networks.
The use of location and time stamp was proposed by [13], where the location is obtained from the user phone's to design an authentication method that is suitable for a web-based education system. The location is obtained using a GPS or base station location if GPS is not available. However, this study did not provide security analysis for the proposed method.
A location-based authentication and authorization mechanism using smartphones is proposed in in [14] for mobile transactions. To obtain the location of the user, they proposed the use of a hybrid approach that combines various technologies that are available in modern smartphones, which increases the user- friendliness of their method. Nevertheless, this was has a security limitation related to user privacy.
Improving the authentication of the portable consumer device that is involved in conducting a transaction at a merchant is proposed in [15]. The server acquires the location of the consumer device and the locations of a merchant. If they match, then the consumer device is authenticated. This work did not provide security analysis for the proposed method.

2.3. Multifactor authentication

A multi-factor authentication that uses the knowledge factor with the location factor is proposed in [14]. The proposed approach uses a combination of various servers and applications, which renders this approach costly in addition to privacy limitations related to user location.
A two-factor authentication that uses biometrics together with a password is proposed in [16]. As discussed previously, the combination of biometrics and knowledge affects the usability of the system, which is an important aspect that impacts the success of smartphone applications.
A multi-factor authentication that uses user biometrics, a user password in addition to location and time is proposed in [17]. This assures secure-live communication, but at the cost of usability, which is an important feature for any authentication scheme.
In this paper, the proposed scheme incorporates only biometrics with location-based authentication, which increases the usability while still gaining all the security benefits.

3. Architecture design

The proposed scheme functions in three phases: setting up, registration and signing in, as shown in Fig1.

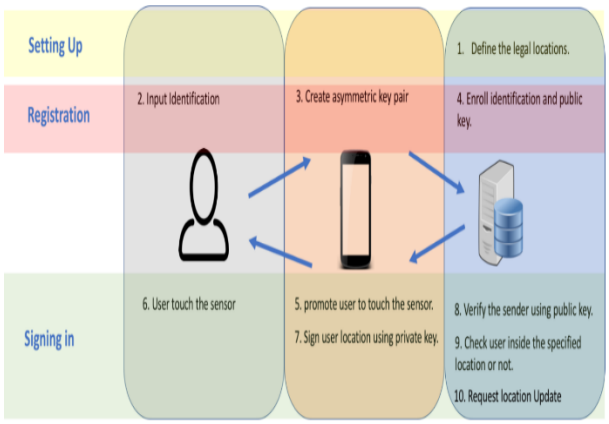


Fig. 1: Architecture Design.

In the setting up phase, the system admin will configure the legal locations of the user. For instance, if the scheme is designed for an air traffic control system, the system admin will define and specify the airport area as a polygon consisting of points designated as latitudes and longitudes.

This is followed by the registration phase in which the user inputs his/her identification and clicks register. Instantly, a pair of asymmetric keys will be created. The private key will be stored in the user's smartphone and the public key will be transmitted to the backend and saved in the system for future use in the verification process. In the previous example, the users should be airport staff.

Finally, the logging in step involves a prompt by the application for the user to touch the sensor; if the scanned fingerprint matches that already stored in the device, user location will be obtained using WI-FI, Network, or GPS in any combination available. This information will be signed using the private key which can only be retrieved if the fingerprint is matched. The signed information is then transmitted to the backend where the system will verify the sender using the public key. Decrypting the received information using the public key will enhance the possibility of determining user location and verifying whether it is within the legal locations defined in Step 1 or not. The authentication will be successful if it is within the legal location, otherwise it will fail. In the previous example, the system will decrypt the received information using the employee public key registered in the system; if the decryption is successful, the system will confirm that the received location is within the airport before authenticating the employee; however, authentication will fail if the location is outside the airport.

4. Prototype implementation

The prototype has been implemented in four stages: setting up, registration, signing in and finishing up.

4.1. Setting up

- Define the legal location

The legal location is defined by establishing several location points (latitudes and longitudes) to create a polygon as shown in Fig2. This is defined as the specific user region which can be stored in the user device or in the server. In this scheme, the information will be stored in the server for flexibility, to allow the application administrator to change the location or adjust the latitudes and longitudes. This will not be possible if the location is stored in the user device which is inconvenient, as it will require all users to update their app to obtain the new location.

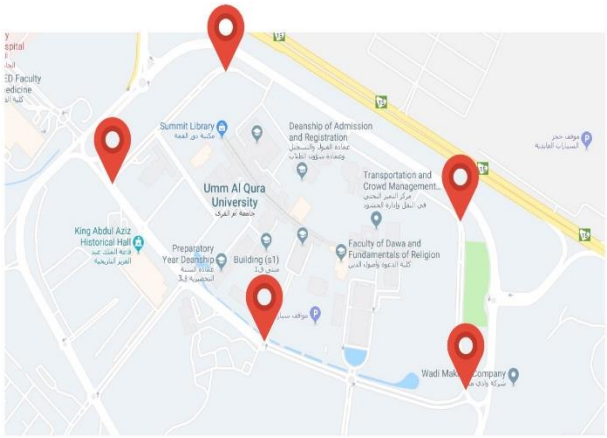


Fig. 2: Defining the Legal Location.

4.2. Registration

- Identification

Identification could take the form of a social security number (SSN), ID number, email, Phone number or other forms of ID. In this prototype, a user name was chosen as identification for convenience

- Create asymmetric key

A pair of public and private keys were created using Elliptic Curve cryptography. The key size is 256 as recommended by NIST. Elliptic-Curve was selected because it uses a smaller key size compared with other asymmetric cryptography techniques, and requires less storage and less processing power which are limited on a smartphone. The asymmetric keys are generated using Key Pair Generator, an engine with the capability to generate the private key and its related public key. This engine will also be used to initialise the Elliptic Curve algorithm.

- Enroll identification and public key

Once the key pair has been created, the private key will be stored in the smartphone KeyStore, a secure repository used to store cryptographic keys, X.509 certificate chains, and trusted certificate [18]. The public key will be transmitted with the identification to the backend for future use, to verify that the authentication request is made by the authorised user. Fig3 shows an example of 256 EEC public key.



Fig. 3: An Example of Public Key.

4.3. Signing in

- Scanning the user fingerprint

After the user clicks on the "sign up" button, the scheme will prompt the user to touch the fingerprint sensor. If it corresponds to the stored fingerprint, the scheme will continue. If there is no match, an error message will be displayed.

The fingerprint will neither be stored in the system nor transmitted; it will only be utilised to authorise the use of the private key. Also, the private key will not be retrieved unless the user is authenticated with a registered fingerprint.

- Determine user location

Once a fingerprint match has been confirmed, the smartphone location will be determined using any available combination of Wi-Fi, cellular networks and/or GPS.

- Sign the information

The obtained location will be signed with the user private key and sent to the backend with a nonce, which is a number that is used to prevent replay attacks by assuring that it is unique and can be used only once. In Fig4 shows an example of the encrypted message.

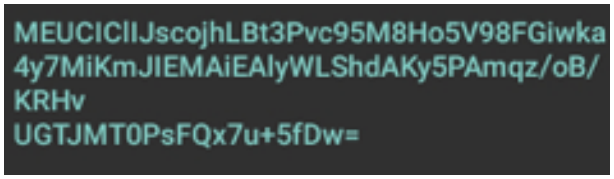


Fig. 4: An Example of the Encrypted Message.

4.4. Finishing up

- Verification

After the signed message is received at the backend, it will be verified using the public key enrolled Previously.

- Checking user is inside a legal location

The public key will be obtained by decrypting the received message using a registered public key. Then, the obtained location will be verified to determine that it is within a legal location. If it is confirmed, the authentication will be successful as shown in Fig5, otherwise it will fail as shown in Fig6.



Fig.5: Authentication Successful.

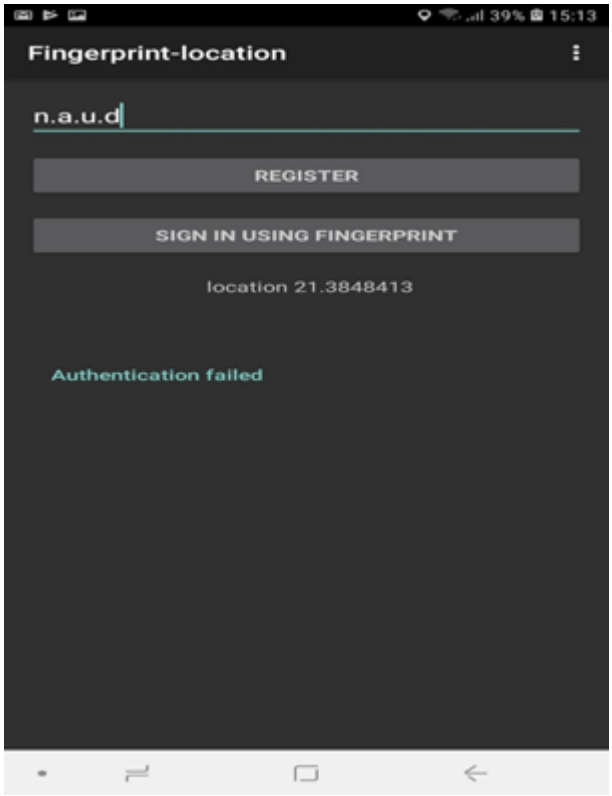


Fig.6: Authentication Fail.

- Request location Update

The scheme will request a user location update every 300 seconds and/or every metre. This configuration option can be adjusted to meet system requirements.

5. Evaluation

we will evaluate the scheme using using the usability-deployability-security (UDS) evaluation framework that is introduced by [19]. It has been widely used to evaluate different authentication schemes such as: ObPwd [20], BlindLogin [21], SV-2FA [22], ZEBRA [23] and QuickAuth [24]. This framework defines 25 benefits covering three evaluation dimensions: Usability includes 8 benefits, Deployability includes 6 benefits and Security includes 11 benefits. Each scheme will be rated as: Offers the benefit, Does not offer the benefit,

Almost offers the benefit, which is indicated by the Quasi- prefix. And Since we are evaluating a scheme for a smartphone app, we will replace Browser-Compatible to Platform Independence for a more accurate description of the benefit.

5.1. Evaluation of the proposed scheme

The proposed scheme is evaluated in detail below; a summary is provided in Table 1.

5.1.1. Usability dimension

- Memory wise-Effortless:

The scheme offers Memorywise-Effortless because it relies on the fingerprint and the location of the user. Therefore, it is not necessary for the user to remember any secret information.

- Scalable-for-Users:
- The scheme also offers Scalable-for-Users. Regardless of the number of accounts the user has, the burden will not increase on the user.
- Nothing-to-Carry:

The scheme also offers Nothing to Carry. Since the scheme is designed for smartphone app authentication, by default the user will be required to carry a smartphone and no other device is necessary.

- Physically-Effortless

The authentication scheme does not offer Physically Effortless, because users are required to be in a specific location in addition to the effort of touching the fingerprint sensor.

- Easy-to-Learn

The scheme also offers easy to learn, since the only requirement is that users learn how to touch the sensor, which is an easy action to recall. Furthermore, the location factor is calculated without user interference.

- Efficient-to-Use

The time the user spends for each authentication is approximately 2,500 ms, which is acceptable. Additionally, the time required for the registration is 1,500 ms, which is also acceptable.

- Infrequent-Errors

The authentication scheme offers Quasi Infrequent-Errors because the false recognition rate is 6%.

- Easy-Recovery-from-Loss

The authentication scheme does not offer Easy-Recovery-from-Loss; if a user is unable to provide his/her fingerprint for any reason, there is no alternate scheme that can authenticate the user.

5.1.2. Deploy ability dimension

- Accessible: Although there were some failures in terms of registering fingerprints, there have been significant developments in newly released devices. Therefore, the scheme can be considered to offer accessibility.
- Negligible-Cost-per-User: The scheme is Negligible-Cost-per-User, since the costs per user are minimal.
- Server-Compatible: The scheme offers Server-Compatibility, since no changes are required from the server end.
- Platform-Independence: The scheme offers Platform-Independence, since no changes are required from the client end.
- Mature: The scheme is not Mature, since it has not been implemented beyond the context of this research.
- Non-Proprietary: The scheme is Non-Proprietary, since anyone can use or implement this scheme.

5.1.3. Security dimension

- Resilient-to-Physical-Observation:

Although location authentication does not offer Resilient-to-Physical-Observation, because of the biometrics, the scheme does offer Resilient-to-Physical-Observation since the user cannot be impersonated if someone observes him/her authenticate one or more times.

- Resilient-to-Targeted-Impersonation

The impersonator can break the location authentication by Physical-Observation, the fingerprint can be lifted from glass surfaces with gelatine-like substances and a legitimate user's smartphone can be stolen by an attacker. Thus, the system is Quasi Resilient-to-Targeted-Impersonation

- Resilient-to-Throttled-Guessing

The scheme offers Resilient-to-Throttled-Guessing, since an attacker cannot successfully guess the private key and the location of the user.

- Resilient-to-Unthrottled-Guessing

Elliptic-Curve cryptography with key size is 256 have not been broken yet. Therefore, an attacker can not guess the private key of the user.

- Resilient-to-Internal-Observation

The scheme is Quasi-Resilient-to-Internal-Observation, since any malware must infect the fingerprint and the location factors in order for an attack to be successful.

- Resilient-to-Leaks-from-Other-Verifiers

The scheme offers Resilient-to-Leaks-from-Other-Verifiers, because even if a user's fingerprint has been leaked from other verifiers, it will not affect the fingerprint authentication factor since the stored data consists of a pair of a private-public keys that is unique for each user in each verifier. The location cannot be leaked from other verifiers since it differs for each verifier.

- Resilient-to-Phishing

The scheme does offer Resilient-to-Phishing, as the fingerprint and user location can be collected from a simulated verifier, but since the fingerprint authentication is stored as a pair of a private and public keys, it will not provide access to the legitimate system.

- Resilient-to-Theft

The scheme offers Resilient-to-Theft since even if the device has been stolen, it will still require user fingerprint and require the user to be in the defined location to authenticate him self.

- No-Trusted-Third Party

The scheme offers No-Trusted-Third Party, since it does not rely on a trusted third party

- Requiring-Explicit-Consent

The scheme offers Requiring-Explicit-Consent. since it cannot be started without the explicit consent of the user to use both his/her fingerprint and location.

- Unlikable

The scheme offers Unlinkable since a public key and verifier are stored in the verifier that are unique for each verifier.

Table 1: Table Summarizing the Evaluation of the Proposed Scheme

Benefit	Offer	Not offer	Almost offer
Memorywise-Effortless	•		
Scalable-for-Users	•		
Nothing-to-Carry	•		
Physically Effortless		•	
Easy-to-Learn	•		
Efficient-to-Use	•		
Infrequent-Errors			•
Easy-Recovery-from-Loss		•	
Accessible	•		
Negligible-Cost-per-User	•		
Server-Compatible	•		
Platform independence	•		
Mature		•	
Non-Proprietary	•		
Resilient-to-Physical-Observation	•		
Resilient-to-Targeted-Impersonation			•
Resilient-to-Throttled-Guessing	•		
Resilient-to-Unthrottled-Guessing	•		
Resilient-to-Internal-Observation			•
Resilient-to-Leaks-from-Other-Verifiers	•		
Resilient-to-Phishing	•		
Resilient-to-Theft	•		
No-Trusted-Third-Party	•		
Requiring-Explicit-Consent	•		
Unlinkable	•		

5.2. Evaluation of other smartphone schemes

we will evaluate various smartphone authentication scheme in detailed , and a summary is provided in Table 2 with references to relevant methods.

5.2.1. Text-based authentication schemes

Text-based authentication schemes belong to the first category (Something you know), and this method is considered to be the most popular and commonly used. Numerous online services use this scheme due to its simplicity, ease and cost-effectiveness. Text-based authentication schemes include: Personal Identification Number (PIN), Password and Pseudo Pressure [25].

5.2.1.1. Usability

In this category, three different methods are evaluated: PIN, Password and Pseudo Pressure. They are not Memorywise-Effortless nor Scalable-for-Users as it is necessary to remember a password for each system. They are also not Physically-Effortless, as user must type their password/PIN. Users are not required to carry anything, and they are Easy-to-Learn and Easy-Recovery-from-Loss, as they can be easily reset. PIN password are also Efficient-to-Use and they are Quasi Infrequent-Errors, while Pseudo Pressure is not, as users find this method to be slower and errors frequently occur.

5.2.1.2. Deploy ability

They are highly deployable since they offer Accessibility, Negligible-Cost-per-User, Server-Compatibility, Platform-Independence and are Non-Proprietary. PIN and password are also mature, since they are the most used schemes for authentication.

5.2.1.3. Security

On the other hand, they offer poor security, since they are not Resilient-to-Physical-Observation as it is possible to acquire someone's PIN or password by observing or recording them type. They are Quasi Resilient-to-Targeted-Impersonation since most people choose a password related to their personal life. They are not Resilient-to-Throttled-Guessing Resilient-to-Unthrottled-Guessing, Resilient-to-Internal-Observation, or Resilient-to-Leaks-from-Other-Verifiers as even a hashed password can be cracked, and they are not Resilient-to-Phishing.
Nevertheless, they are Resilient-to-Theft, No-Trusted-Third-Party, Requiring-Explicit-Consent, and Unlinkable.

5.2.2. Graphical based category

Graphical-Based Passwords also belong to the first category (Something you know). In graphical-based passwords, the secret is shared as a graphic rather than text. The idea behind this is based on psychological research that shows that humans can recognise pictures better than text. Graphical-Based Password techniques are classified into two categories: Recall based techniques and Recognition based technique. Example of Recall based techniques are Pattern lock [26], Draw A Secret (DAS) [27], Syukri [28] Pass doodle [29] Blonder [30], Pass point [31] and Cued Click points [32]. Example of Recognition based techniques Passface [33], Déjà vu [34] and Triangle Scheme [35].

5.2.2.1. Usability

The Graphical based category is not Memorywise-Effortless, Scalable-for-Users, or Physically-Effortless. They do not require users to carry anything, and are Easy-to-Learn and Easy-to-Recovery-from-Loss. They are Quasi Efficient-to-Use and Quasi Infrequent-Errors, as this is significantly dependent on the chosen method.

5.2.2.2. Deploy ability

They are not accessible, as they cannot be used by blind people, and nor are they Server-Compatible. However, they are Negligible-Cost-per-User, Platform-Independence, and Non-Proprietary. They are Quasi Mature for a recall-based scheme, because it been used on one android version.

5.2.2.3. Security

They are not Resilient-to-Physical-Observation, Resilient-to-Throttled-Guessing, or Resilient-to-Unthrottled-Guessing, as it is easier for an attacker to predict a user password in graphic based then in the text based. They are not Resilient-to-Internal-Observation or Resilient-to-Phishing. They are Resilient-to-Targeted-Impersonation, Resilient-to-Theft, No-Trusted-Third-Party, Requiring-Explicit-Consent, Unlinkable. Additionally, Quasi Resilient-to-Leaks-from-Other-Verifiers depends on the specific scheme.

5.2.3. Physical biometrics-based category

Physical based biometrics schemes employ the unique physical characteristics of humans to identify users. This includes Fingerprint Recognition [7] [8], Iris Recognition [36], Face Recognition [37], and Palm Vein Recognition [38].

5.2.3.1. Usability

They are Memorywise-Effortless, Scalable-for-Users, Nothing-to-Carry, Easy-to-Learn , Efficient-to-Use.
Voice and Face recognition are Quasi Efficient because the registration phase can take longer and Infrequent-Errors depending on the implemented methods, However, they are not Easy-Recovery-from-Loss.

5.2.3.2. Deploy ability

They are Accessible since almost everyone now owns them, Negligible-Cost-per-User, and Server-Compatible because the recognition of the biometrics is performed by the technology contained within the smartphone. They are also Platform-Independent and Non-Proprietary except for the Palm Vein, as the proprietary status is not clear. The maturity varies depending on the method with fingerprint as the most mature.

5.2.3.3. Security

They are Resilient-to-Physical-Observation except for iris and face, because they can be photographed, and voice recognition, because it can be recorded.
In this scheme, iris recognition is granted quasi - Resilient-to-Physical-Observation because it can be difficult to take a photograph of the iris, although this is not the case with the face as it is relatively to take a photograph of someone without their knowledge. They are Resilient-to-Targeted-Impersonation except for fingerprint, since it can be lifted from a surface touched by the user and voice recognition because someone's voice can be impersonated to an acceptable degree. They are Resilient-to-Throttled-Guessing and Resilient-to-Unthrottled-Guessing, but not Resilient-to-Phishing or Resilient-to-Internal-Observation as an attacker can capture a reading from the biometric reader and use it to gain access. They are Resilient-to-Theft, No-Trusted-Third-Party, Requiring-Explicit-Consent, and Unlinkable. Quasi Resilient-to-Leaks-from-Other-Verifiers and Quasi unlinkable were granted because there are some privacy-friendly implantations.

5.2.4. Behavioural biometrics-based category

Behavioural-based schemes identify users based on their pattern of doing something. This includes Brain Wave-Based Authentication [39], Recognition of Gestures [40], Keystroke dynamics based Authentication [41] and Context-based Authentication [42] [43].

5.2.4.1. Usability

They are Memorywise-Effortless, Scalable-for-Users, Nothing-to-Carry, and Easy-to-Learn except for Gesture Recognition as it requires user training. They are not Physically-Effortless except for context based, where the date will be captured from the user smartphone without the requirement for any physical effort. It is Efficient-to-Use for Context Based, and Quasi for the Keystroke-dynamics as it depends on the number of strokes that will be analysed and not clear for Gesture Recognition. It is not Infrequent-Errors as the behaviour of the user is effected by his/her situation and mood. It is not Easy-Recovery-from-Loss.

5.2.4.2. Deploy ability

They are Accessible, Negligible-Cost-per-User, Server-Compatible, Platform-Independence and Non-Proprietary; however, they are not Mature.

5.2.4.3. Security

Gesture Recognition and Keystroke-dynamics are not Resilient-to-Physical-Observation and are Quasi Resilient-to-Targeted-Impersonation as the behaviour can be recorded, learned and then replicated. However, it is Resilient-to-Physical-Observation and Resilient-to-Targeted-Impersonation in the Context based. All of them are Resilient-to-Throttled-Guessing, Resilient-to-Unthrottled-Guessing, Resilient-to-Theft, No-Trusted-Third-Party, Requiring-Explicit-Consent and Unlinkable. They are not Resilient-to-Internal-Observation, Resilient-to-Leaks-from-Other-Verifiers and Resilient-to-Phishing.

5.2.5. One-time password

An OTP is a password that is used once and will become invalid after the login session ends or after a specific period of time. The OTP is delivered via text message (SMS), email, mobile app or specific tokens such as SecurID [44]. The security of OTP relies heavily on the medium used for delivery. For example, if it is delivered via SMS, then the security of the mechanism will be dependent on the security of the cellular network.It solves many of the problems associated with traditional passwords, but at the cost of usability.

5.2.5.1. Usability

OTP is Memorywise-Effortless, Scalable-for-Users, and Nothing-to-Carry since a user's email and SMS can be accessed from their smartphone. It is Easy to Learn, but it is not Efficient-to-Use or Physically-Effortless because it will require users to enter their identification in the application, wait a certain period until receiving the code, acquire the code from the other medium, and then return to the application in order to enter the code. It is quasi Infrequent-Errors because of the occasional errors that can occur when entering the OTP. It is also not Easy-Recovery-from-Loss; for example, if the derived medium is SMS and the user loses the SIM card, he/she will not be able to login to the account until he/she has a new SIM card with the same number.

5.2.5.2. Deploy ability

It is Quasi Accessible since the other medium must be available to the user, and Quasi Negligible-Cost-per-User depending on the delivered medium - if it is SMS, there will be cost incurrent when sending an SMS to each user. It is not Server-Compatible as it must be changed depending on the verifier medium. It is Platform Independence, Mature and Non-Proprietary.

5.2.5.3. Security

It is Resilient-to-Physical-Observation, Resilient-to-Targeted-Impersonation, Resilient-to-Throttled-Guessing, Resilient-to-Unthrottled-Guessing, Resilient-to-Leaks-from-Other-Verifiers, and Resilient-to-Phishing because it changes every time and is only available for a short period. It is Quasi Resilient-to-Internal-Observation because an attacker might be capable of intercepting the code and using it before the legitimate user. It is not Resilient-to-Theft as if the smartphone was stolen and not locked, if the notifications are enabled, this could present a problem. It is No-Trusted-Third-Party since it is completely dependent on the trust of the medium. It does Require-Explicit-Consent and it is Unlinkable.

5.2.6. Google 2-step

This technique employs 2-factor authentication: usually the password and OTP that will be sent to the user over text, voice call, mobile app or Security Token. This technique solves the usability issue in OTP by only using the second factor if the user attempts to sign in from a new device. Otherwise, the user can sign in with the traditional password [45].

5.2.6.1. Usability

It is not Memorywise-Effortless nor Scalable-for-Users since users must still remember their password. However, it is Nothing-to-Carry and Easy-to-Learn. It is not Physically-Effortless, Efficient-to-Use, and Quasi Infrequent-Errors because of the OTP step. It is Quasi Easy-Recovery-from-Loss because users must enable backups verifier to the first one.

5.2.6.2. Deploy ability

It is Quasi Accessible, Quasi Negligible-Cost-per-User, Not Server-Compatible, Platform independence, similar to OTP. It is Mature, but it is Non-Proprietary because it is the intellectual property of Google.

5.2.6.3. Security

It is Quasi Resilient-to-Physical-Observation and Quasi Resilient-to-Targeted-Impersonation because an attacker can observe the password and use it when the OTP is not asked. It is Resilient-to-Throttled-Guessing, Resilient-to-Unthrottled-Guessing, Resilient-to-Leaks-from-Other-Verifiers, and Resilient-to-Phishing because of OTP. It is Resilient-to-Theft because of the password requirement. It is Require-Explicit-Consent, No-Trusted-Third-Party and Unlinkable since it is only used by Google.

5.2.7. Password and location based

This technique uses the current location of a user as a second layer to the conventional password. These location-based techniques have different implementations and setups based on the system requirements [14].
The main advantage of this technique is that it prevents attacks launched by stolen devices, or attacks launched from remote locations using stolen passwords.

5.2.7.1. Usability

It is not Memorywise-Effortless, Scalable-for-Users, Physically-Effortless, and Quasi Infrequent-Errors because of the password factor. It is Quasi Easy-Recovery-from-Loss depending on the implementation of the location factor. It is Easy-to-Learn and Efficient-to-Use.

5.2.7.2. Deploy ability

It is Accessible, Negligible-Cost-per-User, Server-Compatible, Platform-Independence and Non-Proprietary. It is not Mature since it has not been deployed on a large scale.

5.2.7.3. Security

It is neither Resilient-to-Physical-Observation nor Resilient-to-Targeted-Impersonation, because both the password and location can be observed. It is Resilient-to-Throttled-Guessing because of the password and Resilient-to-Unthrottled-Guessing as a result of the location. It is not Resilient-to-Internal-Observation or Resilient-to-Phishing because both the password and location can be captured. It is Quasi Resilient-to-Leaks-from-Other-Verifiers and Quasi Unlinkable because it is dependent on the implantation of the scheme. It is Resilient-to-Theft and No-Trusted-Third-Party. It is Quasi Requiring-Explicit-Consent because the user's location can be captured without his/her knowledge.

5.2.8. Loc Biometrics

This technique employs 4 authentication factors, knowledge, biometrics, location and timestamp to assures secure-live communication. but it has some limitation related to usability [17].

5.2.8.1. Usability

The scheme is not Memorywise-Effortless, Scalable-for-Users, or Physically-Effortless because of the password factor.
It requires Nothing-to-Carry and is Easy-to-Learn. It is not Physically-Effortless nor Not Efficient-to-Use since users are required to type their password and then scan their fingerprint. It is Not Infrequent-Errors because unintentional errors can occur when typing the password, in scanning the fingerprint and in extracting the location. It is not Easy-Recovery-from-Loss since the scheme does not propose any form of recovery solution.

5.2.8.2. Deploy ability

It is Accessible, Negligible-Cost-per-User, Platform-Independence and Non-Proprietary. However, it is not Server-Compatible as the server must confirm the user location with the cellular network operator. It is also not Mature as it has yet to be deployed.

5.2.8.3. Security

The use of multiple factors has increased the security of the scheme as it is Resilient-to-Physical-Observation, Resilient-to-Throttled-Guessing, Resilient-to-Unthrottled-Guessing. This is because even if a hacker observes, records or guesses the password or/and location, he/she will not be able to replicate the fingerprint. It is Quasi Resilient-to-Targeted-Impersonation as the password can be captured and recorded, while the location can be obtained through physical observation, and the fingerprint can be lifted from glass surfaces with gelatine-like substances. It is Quasi Resilient-to-Internal-Observation because any malware must capture the password, the image of the fingerprint and the location. It is Resilient-to-Leaks-from-Other-Verifiers, Resilient-to-Phishing, Resilient-to-Theft, Requiring-Explicit-Consent and Unlinkable.

Table 2: Comparative Evaluation of the Various Smartphone Authentication Schemes

			Memorywise-Effortless	Scalable-for-Users	Nothing-to-Carry	Physically-Effortless	Easy-to-Learn	Efficient-to-Use	Infrequent-Errors	Easy-Recovery-from-Loss	Accessible	Negligible-Cost-per-User	Server-Compatible	Platform-Independence	Mature	Non-Proprietary	Resilient-to-Physical-Observation	Resilient-to-Targeted-Impersonation	Resilient-to-Throttled-Guessing	Resilient-to-Unthrottled-Guessing	Resilient-to-Internal-Observation	Resilient-to-Leaks-from-Other-Verifiers	Resilient-to-Phishing	Resilient-to-Theft	No-Trusted-Third-Party	Requiring-Explicit-Consent	Unlinkable
	Technique	References	Usability								Deployability						Security										
Secret	Pin/password					●		●	○	●	●	●	●	●	●	●		○							●	●	●	●
Secret	Pseudo Pressure	[25]				●		●			●	●	●	●	●			●		●					●	●	●	●
Graphical based Password	Recall based technique	[26],[27],[28],[29],[30],[31],[32]				●		●	○	○	●		●		●		●		●				○		●	●	●	●
Graphical based Password	Recognition based technique	[33],[34],[35]				●		●	○	○	●		●		●	○	●		●				○		●	●	●	●
Physical Biometric	Finger Print	[7][8]	●	●	●	●	●	●	○	○		●	●	●	●	●	●	●	●	●	●		○		●	●	●	○
Physical Biometric	Iris Recognition	[36]	●	●	●	●	●	●	●			●	●	●	●	○	●	○	●	●	●		○		●	●	●	○
Physical Biometric	Voice recognition	[33] [34]	●	●	●	●	●	●	○			●	●	●	●	○	●			●	●		○		●	●	○	○
Physical Biometric	Face Recognition	[37]	●	●	●	●	●	●	○	●		●	●	●	●	○	●		●	●	●		○		●	●	○	○
Physical Biometric	Palm Vein Authentication System	[38]	●	●	●	●	●	●	●	●		●	●	●	●	-	●	●	●	●					●	●	●	●
Physical and Behavioral Biometric	Brain Wave Based Authentication	[39]				●				-		●					●	●	●	●					●	●	●	●
Behavioral Biometric	Gestures Recognition	[40]	●	●	●			-				●	●	●			●		○	●	●				●	●	●	●
Behavioral Biometric	Keystroke-dynamics	[41]	●	●	●		●	○				●	●	●			●		○	●	●				●	●	●	●
Behavioral Biometric	Context Based Authentication System	[42],[43]	●	●	●	●	●	●				●	●	●			●	●	●	●	●				●	●	●	●
MFA	One time password	[44]	●	●	●		●		○			○	○		●	●	○	●	●	●	●	○	●	●			●	●
MFA	Google 2-Step	[45]				●		●	○	○	○	○			●	●		○	○	●	●		●	●	●	●	●	●
MFA	Password and Location based	[14]				●		●	●	○	○	●	●	●	●		●			●	●		○		●	●	○	○
MFA	LocBiometrics	[17]				●		●				●	●		●		●	●	○	●	●	○	●	●	●		●	●
MFA	This scheme		●	●	●		●	●	○			●	●	●	●		●	●	○	●	●	○	●	●	●	●	●	●

5.3. Comparative analysis

As can be seen from Table 2, MFA schemes offer more security benefits in comparison to those that only employ single factors, albeit at the cost of usability, or at the cost of deployability such as OTP or in most schemes both, such as Google 2-Step, password and location, and LocBiometrics.
This scheme offers the most benefits in comparison to other MFAs. For example, the OTO offers less security benefits, less deployability benefits with the same number of usability benefits because of the dependence on another medium. Furthermore, Google 2-Step and Password and Location offer less security benefits, less deployability, and less usability benefits. LocBiometrics also offers less usability and deployability benefits because it uses a third factor (passwords). It also offer less security benefits because it depends on the Trusted of Third Party, since it requires the location information from the BS they are associated with.

5.4. Security limitation

In the proposed scheme there is a security limitation in relation to users' privacy. The user location will be accessible at the backend after decrypting the received message using the public key, and since user location is particularly sensitive information, in the future, various measures should be implemented to improve user privacy.

5.5. Performance limitation

In the proposed scheme, there is a performance limitation that materialises in the worst-case scenario where all position technologies are unavailable. However, the effects of this limitation are relatively trivial and will be diminished as the Wi-Fi infrastructure grows and the network and GPS positioning technology improve.

6. Conclusions and future work

In this section, a conclusion of the accomplishments made in this thesis will be presented, along with a discussion on potential future enhancements.

6.1. Conclusions

In this paper, the importance of authentication in computer security has been demonstrated, in addition to how a user's location can be beneficial in many smartphone applications. The different technologies that are built into smartphones are capable of providing this information. Then, a multi-factor authentication scheme has been proposed that uses user fingerprint and location information that is suitable for systems based on a fixed site, such as air traffic control or an organisation that needs to restrict access depending on the locations of its different branches.
A prototype of the proposed scheme has been implemented and tested on an Android platform. The results of this implementation have shown that the proposed scheme can increase the security without affecting the usability or requiring any additional costs through the use of security libraries and the components that already exist in modern smartphones. Finally, the scheme has been evaluated using the UDS framework and compared with other smartphone authentication schemes.

6.2. Future work

The work in this paper represents the initial step in exploring the possibility of using biometrics with the location factor in smartphone applications. Hence, there are still numerous opportunities to extend this work further. The following are some of the enhancements that could be pursued to extend and improve on the work presented in this paper;

- Apply different integrations between the location factor and the fingerprint factor.
- Expand the biometrics factor to all forms of biometric authentication that are available in today's smartphones, such as face recognition.
- Add an extension to the scheme to be used as an alternative in the scenario that a user is unable authenticate him/herself using their fingerprint.
- Include additional location positioning techniques such as Bluetooth to find solutions to the performance limitation issues.
- Extend the scheme to make it suitable for authentication with Internet of Things (IoT) devices.
- Explore different techniques that can be used to ensure the privacy of the user, such as symmetric encryption.
- Add some measures that can be taken to resolve the security limitation regarding user privacy mentioned in 5.4

References

[1] M. Looi, Enhanced authentication services for internet systems using mobile networks, in Global Telecommunications Conference, 2001, 2001.
[2] D. E. Denning and P. F. MacDoran, Location-based authentication: Grounding cyberspace for better security, Internet besieged, October 1997, 167-174.
[3] C. Wullems, M. Looi and A. Clark, nhancing the security of internet applications using location: A new model for tamper-resistant GSM location, in Computers and Communication, 2003., July 2003.
[4] D. Dasgupta, A. Roy and A. Nag, Multi-Factor Authentication, Advances in User Authentication, 2017, 185-233. https://doi.org/10.1007/978-3-319-58808-7_5.
[5] N. Ratha, J. Connell, R. M. Bolle and S. Chikkerur, Cancelable Biometrics: A Case Study in Fingerprints, in 18th International Conference on Pattern Recognition (ICPR'06), 2006. https://doi.org/10.1109/ICPR.2006.353.
[6] N. K. Ratha, S. Chikkerur, J. H. Connell and R. M. Bolle, Generating cancelable fingerprint templates, IEEE Transactions on pattern analysis and machine intelligence, 2007, 561-572. https://doi.org/10.1109/TPAMI.2007.1004.
[7] S. Tulyakov, F. Farooq, P. Mansukhani and V. Govindaraju, Symmetric hash functions for secure fingerprint biometric systems, Pattern Recognition Letters, vol. 28, no. 16, 2007, 2427-2436 https://doi.org/10.1016/j.patrec.2007.08.008.

[8] G. Kumar, S. Tulyakov and V. Govindaraju, Combination of symmetric hash functions for secure fingerprint matching. In Pattern Recognition (ICPR), in 20th International Conference, 2010. https://doi.org/10.1109/ICPR.2010.224.

[9] M. Barni, T. Bianchi, D. Catalano, D. R. M., R. Donida Labati, P. Failla and A. Piva, Privacy-preserving fingercode authentication, in In Proceedings of the 12th ACM workshop on Multimedia and security, 2010. https://doi.org/10.1145/1854229.1854270.

[10] Y. Zhang and F. Koushanfar, Robust privacy-preserving fingerprint authentication, in In Hardware Oriented Security and Trust (HOST), 2016 IEEE International Symposium, 2016. https://doi.org/10.1109/HST.2016.7495547.

[11] D. Denning and P. Macdoran, Location-based authentication: Grounding cyberspace for better security, Computer Fraud & Security, 1996. https://doi.org/10.1016/S1361-3723(97)82613-9.

[12] Y. B. L. Cho and M. T. Goodrich, in In Mobile and Ubiquitous Systems: Networking & Services, 2006 Third Annual International Conference, 2006.

[13] H. Takamizawa and K. Kaijiri, A web authentication system using location information from mobile telephones, in n Proceedings of the IASTED International Conference Web-based Education, 2009.

[14] F. Zhang, A. Kondoro and S. Muftic, Location-based authentication and authorization using smart phones, in n Trust, Security and Privacy in Computing and Communications (TrustCom), 2012. https://doi.org/10.1109/TrustCom.2012.198.

[15] A. Hammad and P. Faith, LOCATION BASED AUTHENTICATION, U.S. Patent No. 20,170,286,953. Washington, DC: U.S. Patent and Trademark Office.2017 ،.

[16] S. H. Khan, M. A. Akbar, F. Shahzad, M. Farooq and Z. Khan, Secure biometric template generation for multi-factor authentication, Pattern Recognition, vol. 48, no. 2, 2015, 458-472, https://doi.org/10.1016/j.patcog.2014.08.024.

[17] I. A. Lami, T. Kuseler, H. Al-Assam and S. Jassim, LocBiometrics: Mobile phone based multifactor biometric authentication with time and location assurance., in In Proc. 18th Telecommunications Forum., 2010.

[18] keytool - Key and Certificate Management Tool, Oracle, http:// docs.oracle.com/javase/6/docs/technotes/tools/solaris/keytool.html. [Accessed 11 MAR 2018].

[19] J. Bonneau, C. Herley, P. C. Van Oorschot and F. Stajano, the quest to replace passwords: A framework for comparative evaluation of web authentication schemes, in Security and Privacy (SP), 2012 IEEE Symposium, 2012. https://doi.org/10.1109/SP.2012.44.

[20] M. MANNAN and P. C. VAN OORSCHOT, Passwords for Both Mobile and Desktop Computers.

[21] Ho, Y. L., Bendrissou, B., Azman, A., & Lau, S. H., BlindLogin: A Graphical Authentication System with Support for Blind and Visually Impaired Users on Smartphones., American Journal of Applied Sciences, 2017. https://doi.org/10.3844/ajassp.2017.551.559.

[22] H Fujii and Y Tsuruoka, SV-2FA: Two-factor user authentication with SMS and voiceprint challenge response., in In Internet Technology and Secured Transactions (ICITST), 2013 8th International Conference, 2013. https://doi.org/10.1109/ICITST.2013.6750207.

[23] S. Mare, A. Molina-Markham, C. Cornelius, R. Peterson and D. Kotz, ZEBRA: Zero-Effort Bilateral Recurring Authentication., Companion report, 2014. https://doi.org/10.1109/SP.2014.51.

[24] X. Zhu, S. Yu and Q. Pei, QuickAuth: Two-Factor Quick Authentication Based on Ambient Sound., in In Global Communications Conference (GLOBECOM), 2016. https://doi.org/10.1109/GLOCOM.2016.7842192.

[25] A. S. Arif, A. Mazalek and W. Stuerzlinger, The use of pseudo pressure in authenticating smartphone users., in Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, 2014. https://doi.org/10.4108/icst.mobiquitous.2014.257919.

[26] K. I. Shin, J. S. Park, J. Y. Lee and J. H. Park, Design and implementation of improved authentication system for android smartphone users., in Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference, 2012. https://doi.org/10.1109/WAINA.2012.31.

[27] I. Jermyn, A. Mayer, F. Monrose, M. K. Reoter and A. D. Rubin, The Design and Analysis of Graphical Passwords, in Proceedings of the 8th USENIX Security Symposium, Washington, DC., 2000.

[28] A. F. Syukri, E. Okamoto and M. Mambo, A user identification system using signature written with mouse, in Australasian Conference on Information Security and Privacy, Berlin, Heidelberg., 1998, July. https://doi.org/10.1007/BFb0053751.

[29] C. Varenhorst, M. V. Kleek and L. Rudolph, Passdoodles: A lightweight authentication method., in Research Science Institute., 2004.

[30] S. Wiedenbeck, J. Waters, J. C. Birget, A. Brodskiy and N. Memon, Authentication using graphical passwords: Effects of tolerance and image choice., in Proceedings of the 2005 symposium on Usable privacy and security, 2005, July. https://doi.org/10.1145/1073001.1073002.

[31] A. Bhand, V. Desale, S. Shirke and S. P. Shirke, Enhancement of password authentication system using graphical images, in Information Processing (ICIP), 2015 International Conference, 2015, December. https://doi.org/10.1109/INFOP.2015.7489381.

[32] V. Moraskar, S. Jaikalyani, M. Saiyyed, J. Gurnani and K. Pendke, Cued Click Point Technique for Graphical Password Authentication, International Journal of Computer Science and Mobile Computing 3 (1)., 2014, 166-172.

[33] P. Corporation, The science behind Passfaces, http:// http://www.passfaces.com/enterprise/resources/white_papers.htm. [Accessed 11 8 2018].

[34] R. Dhamija and A. Perrig, "Deja Vu: A User Study. Using Images for Authentication, in Proceedings of the 9th USENIX Security Symposium, August 2000.

[35] L. Sobrado and J. C. Birget, Graphical passwords., The Rutgers Scholar, an electronic Bulletin for undergraduate research., vol. 4, no. 2002, 12-18.

[36] S. Venugopalan and M. Savvides, How to generate spoofed irises from an iris code template., IEEE Transactions on Information Forensics and Security, vol. 6, no. 2, 2011, 385-395. https://doi.org/10.1109/TIFS.2011.2108288.

[37] A. Swaminathan, N. Kumar and M. R. Kumar, Review of Numerous Facial Recognition Techniques in Image Processing., 2014.

[38] C. Brown, Palm vein authentication system launched for mobile devices, http:// nfcworld.com/2017/01/13/349444/palm-vein-authentication-system-launched-mobile-devices/, [Accessed 15 August 2017].

[39] Y. Renard, F. Lotte, G. Gibert, et al, Open VibE: An Open Source Software Platform to design, Test and Use Brain-Computer Interfaces in Real and Virtual Environments, teleoperators and virtual environments, vol. 19, no. 1, 2010, 35-53. https://doi.org/10.1162/pres.19.1.35.

[40] N. Sae-Bae, K. Ahmed, K. Isbister and N. Memon, Biometric-rich gestures: a novel approach to authentication on multi-touch devices., in n Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, May 2012. https://doi.org/10.1145/2207676.2208543.

[41] U. Garg and Y. K. Meena, User authentication using keystroke recognition., in In Proceedings of international conference on advances in computing, New Delhi, 2013. https://doi.org/10.1007/978-81-322-0740-5_17.

[42] J. C. D. Lima, C. C. Rocha and I. Augustin, A Context-Aware Recommendation System to Behavioral Based Authentication in Mobile and Pervasive Environments., in in 2011 IFIP Ninth International Conference on Embedded and Ubiquitous Computing, October 2011. https://doi.org/10.1109/EUC.2011.2.

[43] M. Jakobsson, E. G. Shi, P. and R. Chow, Implicit authentication for mobile devices,"in In Proceedings of the 4th USENIX conference on Hot topics in security., 2009.

[44] N. Haller, C. Metz, P. Nesser and M. Straw, A one-time password system (No. RFC 2289), 1998. https://doi.org/10.17487/rfc2289.

[45] google 2-step Verification, google, http:// google.com/landing/2step/.