THREE-FACTOR AUTHENTICATION USING SMART PHONE

A dissertation submitted for the Degree of Master of Computer Science

R.Kreshan

Index No: 15440403

Supervisor: Dr. T.N.K. De Zoysa

University of Colombo School of Computing

2018

Abstract

The current environment has most of the authentication type. Authenticate remote client is an important of all organizations and systems. Every organization need to protect their system users and customer sercuirty. Time to time hackers advance hacking types, at the same time security protocol also improved such as simple password, two-factor authentication and three-factor authentication. Now a day there is lot of research going on to find best authentication protocol. In common, there are three authentication factors something user knows; Simple password, Something user has; Smart phone, smart card or hardware token and Something user is; Biometric. Development of a protocol with the intention to solve these problems is proposed in this thesis. The objective of the project is to come up with a security protocol that protection with the security. The PKI asymmetric key encryption is used for secure communications along with hashing. For this, a proposed approach is available in the document under methodology and design. It describes the entire flow of the three-factor authentication protocol with the above concepts in track. The scope of this project is to implement this protocol only for point of authenticate remote user. This can be further extended for other scenarios (online user authentication) as well and that will be reserved as future work.

This thesis covers design and implementation of a smart phone based three-factor authentication. Use PKI certificate and private key for secure end to end communication. Smart phone and web application directly connect with security engine, being regulatory compliant are some of the primary aims identified to be achieve Implementation of the proof-of-concept for the designed protocol and production implementation idea realize the primary objects.

Acknowledgement

I would like to express my sincere gratitude to my supervisor, Dr. Kasun De Zoysa, for sharing his experience and guidance in planning the course of this thesis and inspiring me in choosing this topic.

A very big thanks to all the great people who have committed their life for research on these areas and who are in mind and body, always trying to form a better world for others with science, without them I would be in suffering writing this thesis.

I also like to thank my immediate boss Mr. Krishan Jayawardena for providing a good situation, freedom and ability to completion of my research during my work at Epic Lanka (Pvt) Ltd. I also express my kind gratitude to the members of my family for the support they provided throughout. It is also very much significant to pay my gratitude to my colleagues and friends who helped me to make my research a success.

Kreshan Rajendran
University of Colombo School of Computing (UCSC)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS AND ACRONYMS

OTP	One Time Password
SMS	Short Message Service
PKI	Public Key Infrastructure
NFC	Near Field Communication
QR-code	Quick Response Code
MCC	Mobile Cloud Computing
MFA	Multi Factor Authentication
USIM	Universal Subscriber Identity Module
GSM	Global System for Mobile
SIFT	Scale Invariant Feature Transform
SPRF	Speeded-Up Robust Features
DMZ	De Military Zone
CSR	Certificate Sign Request

1 Introduction

1.1 Introduction to the problem

Authenticate the remote client is a common problem in all organization. Every organization need to protect their system users and customer security. To resolve this need the best authentication protocol. In common, there are three authentication factors.

1) Something user knows? Simple password
2) Something user has? Smart phone, smart card or hardware token
3) Something user is? Biometric

The most early used authentication protocol is a simple password, it has more vulnerability. Such as simple password can easily guessable, multiple accounts are using the same password, manually to keep their password in a notebook or asking the web browser to remember their password, by these vulnerabilities using simple tricks attacker can trace the password. Due to these concerns, second-factor authentication plays a major role in this authentication mechanism.

For the second-factor authentication, some organizations start to use hardware token. That account uses different hardware token. The major issue in user and organization is carrying multiple tokens, cost of the hardware token and maintaining the token. For the mobile SMS based second-factor authentication user received OTP through SMS and automated voice call. There are some drawbacks sharing organization metadata with service provider (e.g., dialog) identifies customer phone numbers and then they can find customer personal details with customer virtual account. And some country's (e.g. Iran, Russia, and USA) shown that determined hackers can sometimes hijack the SMS messages meant to find a better remote client authentication protocol.

1.2 Motivation

Smartphones become part of everyone's life. The idea is everyone uses a mobile phone for authentication purpose or financial transactions. There are numbers of authentication mechanisms are using to authenticate remote user. Day by day hacking type is updated; reason for that, there is a big requirement to secure the system users and their private information with latest updated authentication.

If the identifications match, authentication is the main process in which the credentials provided by a security server and system user. When authentication process was done then user is granted authorization for access. The access privileges and favorites arranged for the authorized account depend on the system user permissions, whatever stored locally or on the authentication server. Currently, a number of Internet enabled devices and web systems increases day by day, reliable system authentication is solved to allow secure communication in smart phone applications and other networked environments with multiply application. Each networked device needs strong authentication and also, depend their normally limited activity, these system must be configured for limited permissions access as well. In the Internet of environment scenario, which is increasingly becoming a reality, almost any possible entity may be made and able to interchange data over an internet.

The project motivation is authorized a remote user with secure three-factor authentication using the smartphone. Without integrating third-party SMS service provider. The smartphone now supports all kind of biometric senses, Investigate and finding best biometric for the smartphone with user-friendly manner. The system user uses multiple systems every day each system has their own authentication system to secure their users. If find any secure vulnerable on any of the system authentication, the issue is not fixed in other systems. That is a big benefit to system hackers. So using single authentication protocol has a benefit to fix the issue at single point.

Most websites and users, however, still stuck with the old authentication mechanism as from the web username and password. To resolve authentication problem is will be much more challenging if they are the right result for our projects. The user doesn't like to spend more time to authorization. So introduces a protocol with small time period it check all three authentication factors. Introduces three-factor smartphone app with latest mobile technology such as QR code or NFC, to make more user-friendly.

1.3 Aims and Objectives

- Investigate, design and develop best three-factor authentication protocols for the smartphone. Instead of using two-factor authentications such as hardware token and SMS based OTP transfer.
- Identify best biometric for the third-factor authentication. Investigate some vulnerability in biometric (e.g. Fingerprint can easily obtain without awareness of the owner).

- ❖ Investigate existing authentication protocols. Identify the limitation of the three-factor authentication protocol. Investigate applicable domain. Investigate latest mobile technologies such as NFC or QR-code to make three-factor authentications more users friendly.

1.4 Project scope

Mobile application and the authentication server are the main components of this system and these two systems are included in several subcomponents.

- ❖ Mobile application
  - User portal
  - Biometric implementation
  - Make user-friendly (NFC or QR-code)
- ❖ Security engine
  - Web service for mobile devices
  - Admin panel
  - PKI infrastructure
- ❖ Demo web application
  - Login with security engine

Develop good quality mobile base three-factor authentication protocol between the web server and smartphone. Find a better secure way to communicate OTP. Apply QR code or NFC technique and biometric authentication to the smartphone, to identify best three-factor authentication protocols. To evaluate the project follow use cases and test cases.

1.5 Structure of the thesis

This thesis is structured as follows: Chapter 2 discusses the background of this implementation with related publication on literature. Chapter 3 describes system analysis and design of the proposed solution. Chapter 4 describes the project plan of the system implementation and progress on time plane.

2 Background

2.1 Authentications remote user

The most of the systems remote user authentication now using is two-factor authentication using SMS [1], [2], [3]. It's transferring OTP using SMS service provider.

Use of simple passwords hackers follow some various techniques to hack, such as guessing attack, shoulder surfing attack, dictionary attack, brute force attack, snooping attack, social engineering attack, and some other techniques. To resolve this password issues in banking sectors using two-factor authentications in an online transaction. To square the authentication not only sending OTP through SMS, it implements another layer to sending notification/alert SMS to the customer. When a user tries a wrong password three times or hackers login from an unknown location its sending alert to customer [2]. They are such as banks, governmental applications; healthcare industry, military organization, educational institutions and other organization protect their system users and customers more securely. Improvement in authentication techniques has to consider out feature validation, not today. Now the limitation is biometric authentication leads to the authentication process.

2.2 Soft token application

For the second-factor authentication purpose, instead of using hardware token, smart card or computer base software token, using mobile-based soft token application (OTP generate in a mobile application, e.g. Google authenticator mobile application). This proposed the system to be secure and consists of three processes, they are

1. Software installed on the remote user mobile phone
2. Backend server software
3. GSM modem connected to the server.

In this soft token application based authentication two choices are there operation connection less authentication and SMS based authentication System. Connection less is the mobile phone itself generates OTP without connecting to the remote server. Here the phone performance like hardware token. SMS based authentication is when the first method fails mobile get OTP from the server. The server sends OTP using SMS. The end user has resent that OTP before the time expires. The

drawback of this method is, need to pay telecommunication charges for SMS both the client and server side.

The proposed authentication system uses some techniques to generate secure OTP. OTP generating algorithm use IMEI number, IMSI number (SIM), username, pin and time [3]. These variables must exist on both the mobile phone and the server sides to retrieve the same OTP. This will ensure the correct time synchronization between both sides.

2.3 Biometric authentication

The Greek words “bio” and “metrics” mean “life” and “to measure” respectively. This is where the word biometrics is derived from [4]. The term biometrics in computer science is a basic way of defining the process of authenticates a person with least possibility of error existences or without error. Biometric is accurate and unique characters available in a human body and human behavior.

However since the biometric is not 100% accurate there are more research ongoing process. Because of this, some characters already in use for such purposes are not limited to define and limit the biometrics characters. There are more discoveries and stronger characters which can be derived for more reliably in authentication. Some characters of human biometric, which can be implemented in computer authentication is a fingerprint, hand, eye, face, DNA, signature (hand writing) and voice. Biometric can be categorized into two categories such as Contact Biometric and Contactless Biometric. The contact templates require physically touching the device and Contactless templates do not need to touch any devices, it uses some senses.

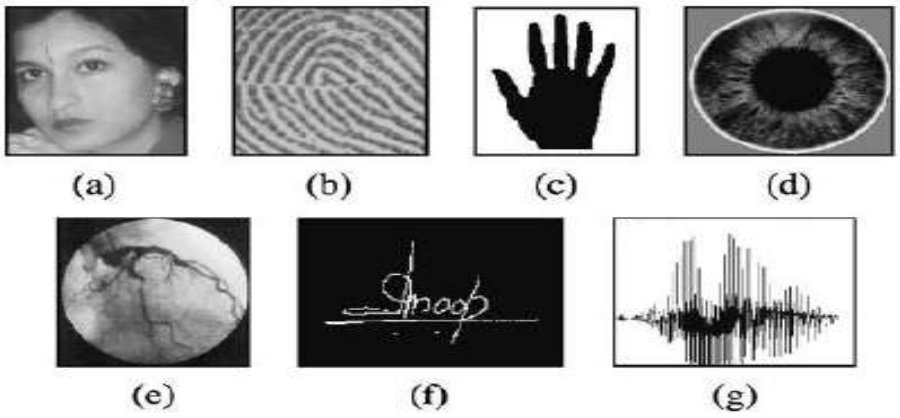


Figure 1 : Biometrics authentication list

When taking the biometric image it can be hugely affected by the nearby factors, such as lighting whilst capturing, the face direction, hair color, facial expression, brightness and also human face changes with age. There is a limitation in developed a complex algorithm. There is no algorithm 100% successfully cover-up human biometric. The Issues of multi biometrics authentication are which involve multiple input data, multiple sources of data collected and multiple protection methods to be implemented. It increases the difficulty of handling multiple SDK to be maintained along with the development

- **Positive in biometric:** Biometric characters are reliable authentication factor and cannot be lost or forgotten easily.
- **Negative in biometric:** It can be obtained easily without awareness of owner (e.g., fingerprint). The biometric characters are totally meaningless.

Biometrics is useful for verifying the person by his behavioral features, Because of high robustness and reliability. The face recognition biometric is the most excellent biometric methods to identify the person, because it has meaning full. When face image was taken from webcam it can be resized and eliminate back ground, then the use of image processing algorithm to face detection module. Then for the verification purpose it uses two inputs image such as Image in database and Image from the webcam.

There are several protocols have biometric authentication, password authentication, and smart-card authentication. There is a researcher designed an authentication protocol which doesn't need password table in the database to authenticate registered users. Instead of that, using smart-card and fingerprints are used to store biometric data for the authentication process [5].

Now we have SIFT and SURF image processing algorithm to get feature extraction more accurately, The algoritham recorgonize features correctly when image Scaled,slidely rotate And noicy image.

2.4 Two-factor authentication

Two-factor authentication (2FA) is often referred as two step authentication process, is a security process in which the user provides two authentication factors to identify they are who they say they are. 2FA can be formed with single-factor authentication which the user provides only one factor such as simple password. One of two factor authentication in market is hardware tokens and SMS based. Hard ware token is supporting various methods of authentication. One popular hardware token is YubiKey and other one is a USB device that supports one-time passwords (OTP) and using

public key encryption and authentication. and another one is SMS based second factor authentication, it's using first factor is normal system password on the system and asking challenge OTP from user ,the OTP send to user SIM card.

2.5 Three-factor authentications

The protocol is authenticating the remote user using three factors such as simple password, smart card, and biometrics. The researcher proposed a protocol to upgrade two-factor authentication to three-factor authentication [5]. The researcher has followed five steps to design three-factor authentication protocol.

- Three-Factor-Initialization
- Three-Factor-Registration
- Three-Factor-Login-Authentication
- Three-Factor-Password-Changing
- Three-Factor-Biometrics-Changing

The researcher's contribution is introducing a generic framework for three-factor authentication in distributed systems. On the research biometric character is kept from servers to avoid single point failure. Use Fuzzy Extractor methods by two actions, such as probabilistic generation and deterministic reproduction. The researcher limitation is the authentication boundary is defined within smart card.

There are special authentication algorithms are proposed for mobile cloud computing. In mobile cloud computing needs to pooled computation sharing resources and applying more complicated authentication for using different authentication factors [6], when using multi-factor authentication there are some limitation, such as processing power and battery lifetime. Three-factor authentication is three following independent authentication factors, admin can configure how many factor validate for user authentication.

2.5.1 Simple password Authentication

The common problem to the user is the difficulty of memorizing passwords. Because of that, the user uses simple passwords. On this concern, the password can be easily hacked by attackers.

2.5.2 Possession property Authentication

For a particular time period USB Tokens, Smart Cards, and Public-Key Infrastructure to implement this kind of authentication. Need more resources and authentication algorithm consumes more energy.

- Digital signatures created on a SIM card user private key on SIM card.
- Smartphone APN and Android's C2DM/GCM, can be used to provide a real-time response mechanism on a mobile device.
- Soft tokens and hardware tokens (Magnetic stripe cards, Smartcards, Wireless RFID-based tokens, USB tokens and Audio Port tokens).

2.5.3 Biometric-based Authentication

The biometric features are not replaceable, so an attacker can use this malicious opponent to get user secure biometric data. All the mobile devices do not have a all type of biometric reader, to recognize biometric characteristics.

2.5.4 Common Authentication Methods

Authorization is the method of giving individuals access from system information. Based on their identity [7], such as

- Password and PIN authentication
- SMS authentication: SMS is used send a one-time password (OTP).
- Symmetric-key authentication: user shares a secret, unique key with an authentication server. To authorize send a randomly generated message encrypted by the secret key.
- Public-key authentication: In Public-key cryptography a pair of private key and public key. A private key is kept secret by the user, public key shared with the server.
- Digital Signatures authentication: using PKI architecture, encrypt the message with the private key is A digital signature is a digest calculated from a signed document.
- Biometric authentication

Cloud computing permits access to information and computer properties using computational services (Online storage, social networking sites, webmail and online desktop applications) which allows to access software and hardware that are accomplished by a third party at remote locations. Cloud computing network access to a share pool of configurable computing resources (Networks, servers, storage, applications, and services) The researcher authentication technique and proposed

authentication technique with three comparison parameters. Such as chance of success for breaking the authentication system, Single-sign on access of cloud facilities and no of validation factor By using the secret code on SMTP protocol mechanism, the planned authentication technique provides the single sign on access of the cloud facilities provided by the service providers. The user has to provide a top-secret code which is getting on the noted mail id for accessing the exact requested service [14].

User can authenticate use properties such as MAC address or IP address; MAC address is authenticating the machine, not the person who is authenticating the users who usually have access to their accounts from a regular set of machineries. Verification by IP address is positive or not depending on the network from which the access to connected. The three main Cloud services are Infrastructure as a Service (IaaS), Plat form as a Service (PaaS) and Software as a Service (SaaS). Each Cloud service can belong to Cloud deployment models, such as public, private or hybrid. It is related to the security of virtualization technology. Authentication systems can be categorized into two, such as physical character based and behavior based. [15]

The current various attacks such as replay attack, guessing attack, modification attack, and stolen-verifier attack. OTP make it more difficult to gain unauthorized access to controlled resources. The ticket-based onetime password authentication system is more secure than guessing attack and replay attack. The main use of ticket are trusted third party, RC is to generate Ticket to user. The user must use Ticket instead of password in registration process to get the strong confidentiality. The users then will be barred from attacks like brute force attack, phishing, Distributed Denial of Service (DDOS) through password encryption and multifactor authentication via OTP [16].

2.6 Multi-factor Authentication

IBM knowledge center released mobile multi-factor authentication application framework, which is available mobile application play stores; it is supported by the access control component. The application is built on the security access manager SDK; who are the mobile developers, they can use when creating mobile applications [8]. Security access manager SDK provides several predefined authentication policies to enable combinations of the second factor and biometric mechanisms. IBM SDK have the following process

- Authenticator registration: - The IBM Verify application uses the authorization allowance to flow to the perform registration, which is deployed by the user in a browser.

- Authentication method enrollment: - After an authenticator is registered, the user is prompted to enroll authentication methods.
- Configuring Mobile Multi-Factor Authentication: - Follow these steps to configure mobile multi-factor authentication.
- MMFA mapping rule methods: - Customize the authentication belong both method, pre token generation, and post token generation mapping rules.

Multi factor authentication is Password authentication, USIM card authentication, and biometric facial authentication. This authentication scheme has a vulnerable. Middle man attack and replay attack not resolved. This work, android based mobile payment facility using three-factor authentications. In the near future, additional multifactor authentication features and virtual private networking features will keep being developed and integrated. This work offers simple but in a practical method for face recognition, eigenvectors [10]. That recognizes the person by comparing a characteristic of the face to those of identified individual.

The MFA mechanisms independent of one another such that access to one factor does not grant access to any other factor, For example, if the same set of identifications (username/password) is used as an authentication factor and also for ahead access to an e-mail account used for send OTP, and a software certificate (something you have) stored on a personal computer that is protected by the same set of credentials these do not provide independence. The researcher try to use independence of verification factors is often able through physical split-up of the factors. The researcher explain with four scenarios such as 1) An individual uses one set of credentials (passwords) to log in to a device and access a software token stored on the device memory. 2) On second scenario, the each uses one set of credentials (username/password or biometric) to log in to the device. 3) And next scenario individual uses one set of credentials (username/password) to log in into the computer. The connection to the CDE/corporate network needs both the first set of credentials and an OTP generated by software token exist in on a mobile device. 4) On final scenario, the device (smartphone or laptop) should be tough and controlled to guarantee that the multi-factor authentication is properly applied and always performed before initiating the joining to the CDE or corporate network [13].

Researcher proposes multifactor authentication architecture on public Displays. Users authenticate by Gaze-Touch password (knowledge factor) and personal mobile device (possession factor). A grouping of a thermal attack to uncover touch input and an observation attack to

uncover gaze input, or multiple consecutive comments by insiders. Public displays like train stations, airports, and streets. Meanwhile, there is an increasing demand for shows to offer personalized. A further direction for future work is to comprise a third inherence factor. This can be done by scanning the finger print or by face discovery using the front-facing camera. The researcher discussed how thermal and stain attacks are infeasible by design. In the future, people want to evaluate more complex threat models [17].

The idea is randomly selects two of the four stages that are required to authenticate the system. In the first stage the user selects a pattern of boxes from a grid of boxes. In the second stage the user selects five characters out of ten according to a numeric code created at registration. In the third stage the user enters a passcode based on a seed value by using a secret formula installed on the user's smartphone. The fourth stage grants the user with two security questions. In multifactor authentication, two or more elements are used to verify the user's character and the attacker need bypass in some stages. This paper with Soft Token, RFID, QR-Login, and biometric techniques, each has its own advantages and disadvantages. This is a unique idea that we introduced to increase security without forgoing convenience. And they presented a formal mathematical evaluation for stages one and two, and showed the relatively low possibility figures for brute-force and guessing attacks. Implemented and published the web application online and ran a survey to evaluate usability of the authentication system, with very encouraging results [18].

The limitation is to further advance the system by moving the characters' maps into CAPCHA-like arrival to make bot based brute-force attacks more hard.

Any interested campus department with local account space, Active Directory, local systems, and IT services VPN, Application Development Tools, and Data Center Systems. Allows both user pre-registration and Just In Time (JIT) account and device process. Researcher begins using MFA when signing into SSO. UCLA Logon MFA Distribution Summary Single Sign-On, Campus VPN, Campus Wi-Fi, UCLA Logon Active Directory (lab AD authentication) [19].

2.7 Authentication protocols

An authentication protocol is a type of cryptographic protocol specifically designed for transfer data between two entities such as Client to Server and server to server. There is a number of protocols developed day by day, such as Single sign-on, Kerberos, OpenId and OAuth .such protocol details are given below

2.7.1 SSO (single sign-on)

Single sign-on (SSO) is a user authentication service in a single point that permits a user to use the same set of login credentials such as username and password to access multiple applications. The service authenticates the end user for all the applications the user has been given rights to and removes further goads when the user modifications applications during the same session. On the back end server, SSO is helpful for authenticating user activities, as well as SSO server, was maintaining user accounts. Some of SSO service use Kerberos protocol. For example, SSO was enabled a user or system to access multiple computer platforms or application systems after being authenticated just one-time user's identity and authorization data is stored in this centralized setup, which is trusted by all other applications or system.

2.7.2 Kerberos

Kerberos is an authenticating protocol service requests among trusted hosts thru an untrusted network. It was built into all major operating systems such as Microsoft Windows, Apple, FreeBSD, and Linux. It was originally for Project Athena at the Massachusetts Institute of Technology. Kerberos logo was a three-headed dog that guarded the gates of Hades. The three heads represent was a client, server, and Key Distribution Center. The key distribution center is trusted third-party authentication service. The drawbacks are single point failure, all network service which needs a different hostname it use own set of Kerberos keys; Kerberos has strict time requirements, which must be synchronized within configured limits, and Kerberos use DES algorithm can be used in mixture, but it is no lengthier an Internet standard because it was weak algorithm.

2.7.3 OpenID

OpenID gave access to use an existing user account to sign in to multiple websites, without creating new passwords. The user can choose to associate information with user OpenID the data can be shared with the websites what user visited web pages, line name, email or contact number. With OpenID, the user can control what information that can share with the websites. With OpenID user password that is only given to the user to identify and that identity then confirms to

the user to visit the websites. Any website can't see system user password. So users don't need to worry about insecure website identity. Several huge organizations accept OpenID including Google, Facebook, Yahoo, Microsoft, MySpace and many more.

2.7.4 OAuth

OAuth is an open standard for access group, usually used as a way for Internet users to permit system access to their info on other websites but without giving them the passwords. OAuth used by organizations such as Amazon, Google, Facebook, Microsoft, Twitter, and others to permit the users to share information about their accounts with third-party applications. OAuth has 2 major version 1.0 and 2.0, but 2.0 is not backward compatible with 1.0. OAuth is a service that is balancing to and different from OpenID. OAuth started in 2006 when Blaine Cook was developing the Twitter OpenID implementation. The version 2.0 provides specific authorization flows for web applications, desktop applications, and mobile phones.

2.8 Comparison of similar approach

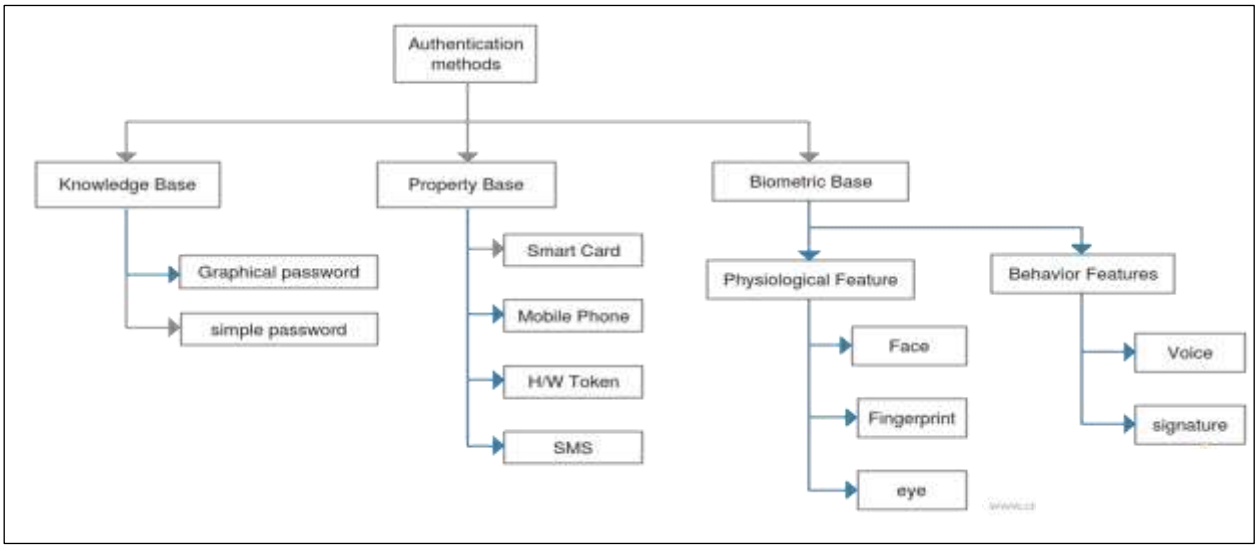


Figure 2 : Classification of authentication methods

Above figure 2 show main categories of authentication types. Follow some examples of those knowledge base, property based and biometric base types.

2.8.1 Authentication factor based comparison

One-factor	Two-factor	Three-factor
Knowledge based	Property based	Biometric based
User friendly high	User friendly medium	User friendly low
Security low	Security medium	Security high
Can be forget	Can be lost	No need to remember and cannot be lost or stolen

2.8.2 Biometric Type comparison

Face	Voice	Finger print
Meaning full	Meaning less	Meaning less
Supported mobile phone high	Supported mobile phone high	Supported mobile phone low
Use SIFT, SURF and Canny edge deduction algorithms		

2.8.3 Other authentication validation techniques

- SMS alert message sending to the customer, because of someone tries to login to the system with wrong password or unknown location.
- Validate customer mobile GPS location and cross-check the GPS location with tower location.
- Using PKI infrastructure (SSL certificate) to secure communication.

2.9 Chapter summary

Here we discuss with an authentication protocol, evaluation system, and supported application with similar research area. Such as OTP, Soft token, user biometric, 2FA, 3FA, Multi factor-authentication and authentication protocols. And also evolving well-established solutions were discussed in details, and discussion similar architecture such as existing authentication protocol. There is a different type of authentication methods such as knowledge base, property base, and biometric base, and here it is compared factor based compilation (single-factor, three-factor and multi-factor) and discussing smartphone supported biometric authentications such as fingerprint, face and voice.

3 Analysis and Design

The proposed system is remote user authentication using a smartphone with three-factor authentication level. The proposed system increases the security and reduces response time, with user-friendly manner. The system is designed smartphone in client-side and backend server. The overall architecture and the functional overview of the design of the system are discussed in this chapter.

3.1 Design Overview

In the proposed solution there are four modules. Those are user mobile application, web service, web portal and service Engine. The mobile application retrieves all three authentication factors and sends to REST Full web service. The web service runs on J2EE web server and it runs on DMZ (Demilitarize zone). Service engine checks the user credential with core database, its run on the local network (military zone). Web service contains security module with PKI certificate

3.2 Design Assumptions

The smartphone has fingerprint sensors or camera to capture biometric, Android OS level store biometric and retrieves the biometric key to share with the web server. When user login to the system on a computer using browser server check the simple password then generate OTP and encrypt by server master key and server private key. Encrypt OTP show in client browser in QR code format. User has been scanning that QR code using his mobile phone from web browser, then retrieved OTP. Then after OTP send to backend server to authenticate, with other authentication factor credential. If authentication is success browser can login success, otherwise fail alert message show in mobile, user can retry.

3.3 System Architecture

The following figure gives high-level system architecture, with main four components. PKI infrastructure used to here to the public key and private key to encryption transaction message packet.

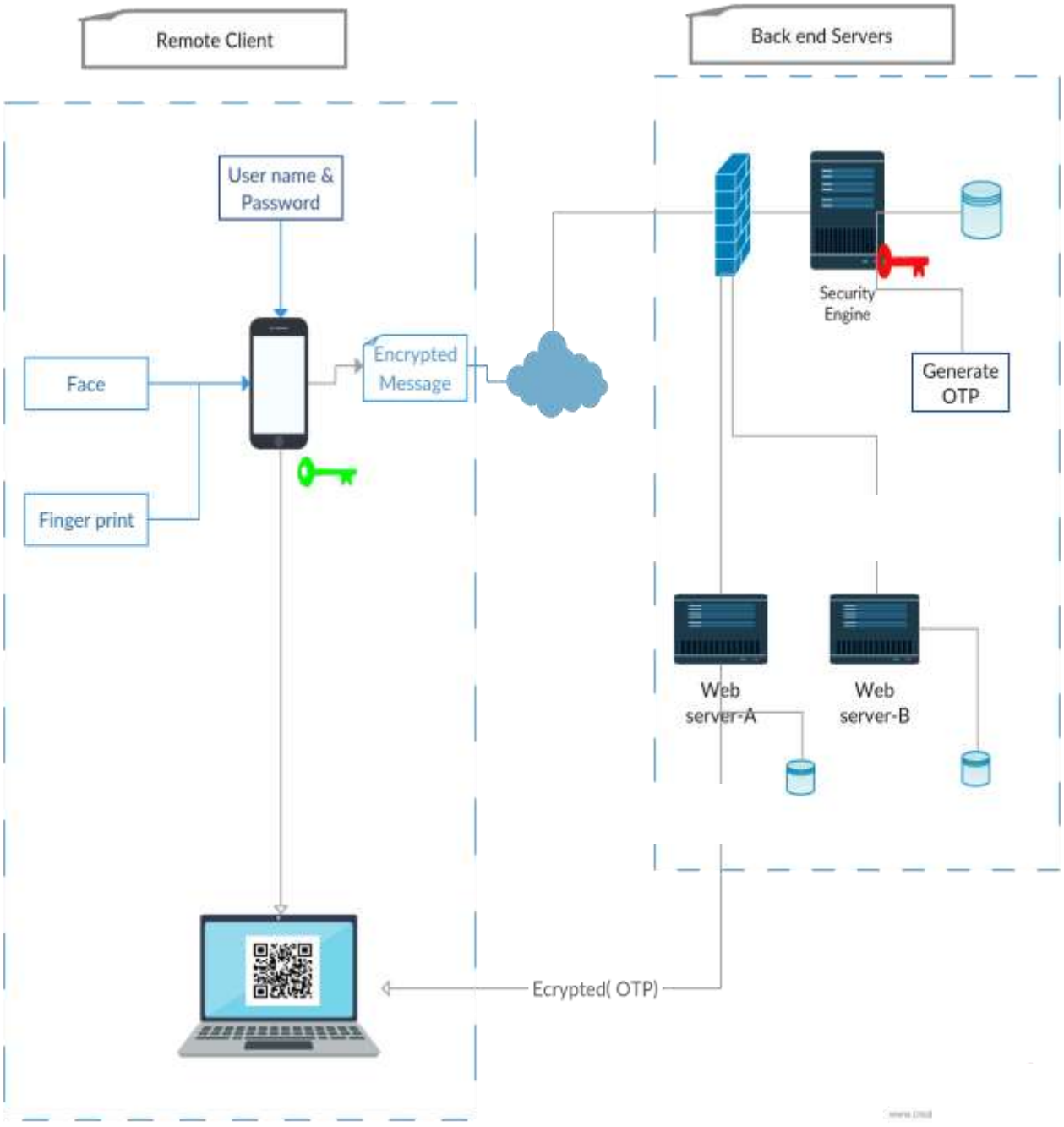


Figure 3 : System architecture

3.4 Functional Overview with sequence diagrams

This section gives the functional overview of the system, which has subcomponents, described. The system consists of a Web service, service engine, web portal, a DBMS and Mobile application with end-user web portal and mobile interface. There are five functional operations conducts with above component. These operations fulfill authentication process. The first operation is initiation process on this process install mobile application and download certificate from the server. Then register the mobile device with authentication server, on the registration process user-selected biometric send to server and generate same master key both mobile and server side. Then login process user authentication is checking in login process. There is a systematic way to handle password change, device change, and biometric change.

Following figures 4, 5, 6, 7, 8, 9 illustrate the functional overview of the system. There are five functional operations conducts with application process

3.4.1 Mobile Initiation

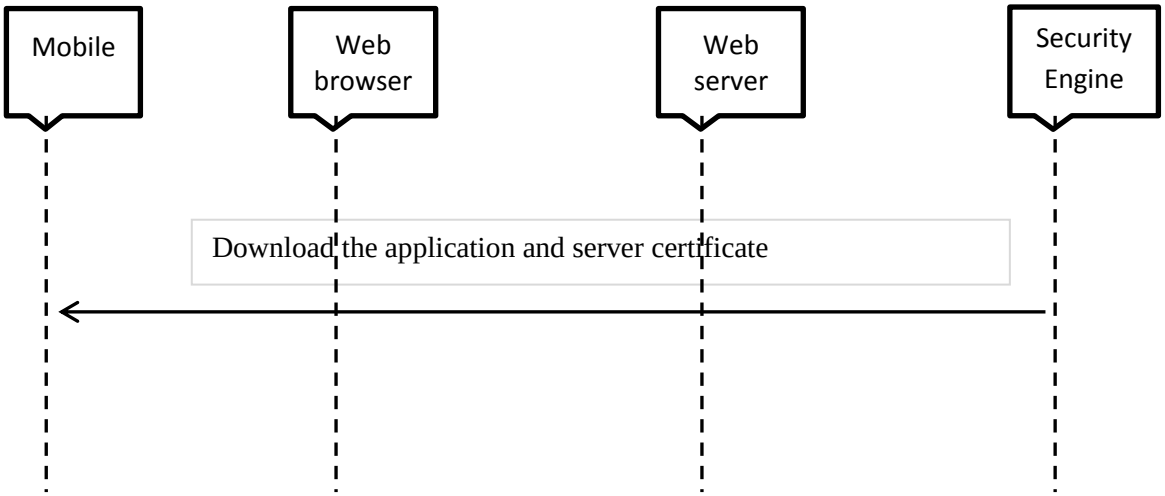


Figure 4 : Mobile initiation download mobile application

First download the application from android play store and install the application to the mobile phone. Server certificate downloaded from security engine server. Use 4 digit pin to start the application, the 4 digit pin stored in SQLite dB in hash (SHA2) format.

There is an option using timeserver service both client and server side. The timeserver synchronizes the same time with in server and mobile device, to block the replay attack.

3.4.2 Mobile Registration

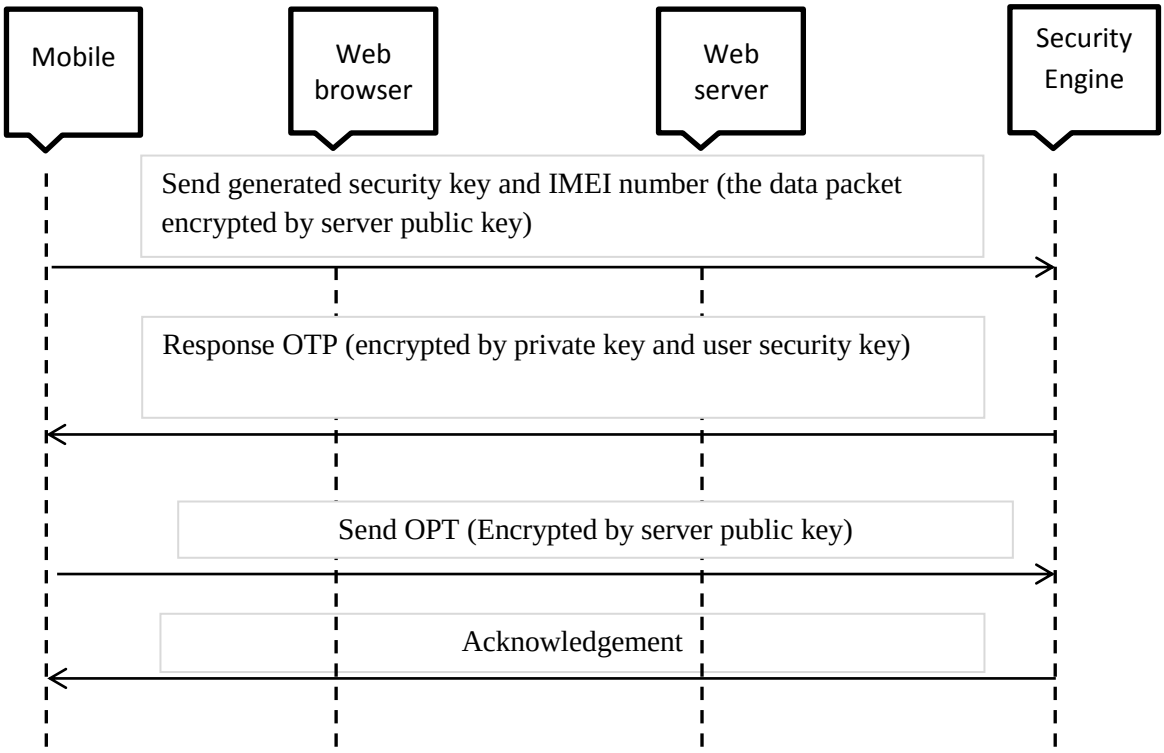


Figure 5 : Register mobile application with security engine

Here mobile phone directly registered with security engine. In that process mobile phone sends all three factor details such as IMEI no, password and biometric key to server. Mobile phone use certificate (server public key) to encrypt transaction packet. Security server stored the user data against mobile IMEI number. 00Multiply application can authentication with same mobile IMEI number or unique key generate by server.

3.4.3 User Add

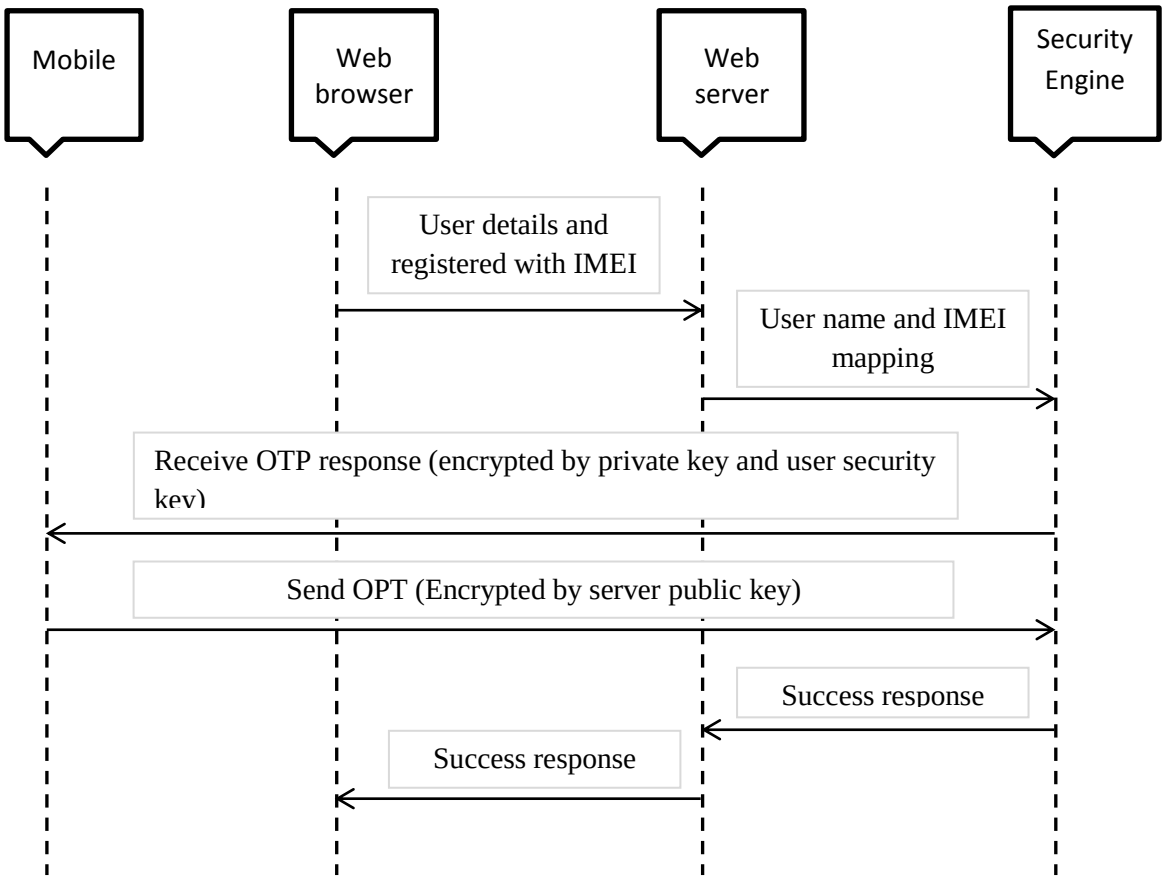


Figure 6 : Web application user add

When web application user needs to three factors authenticate from his mobile phone, then user need to redirect the authenticate part to security server. Security server adds the user with previously registered mobile device. Security server sends the confirmation to smart phone using OTP.

3.4.4 Login

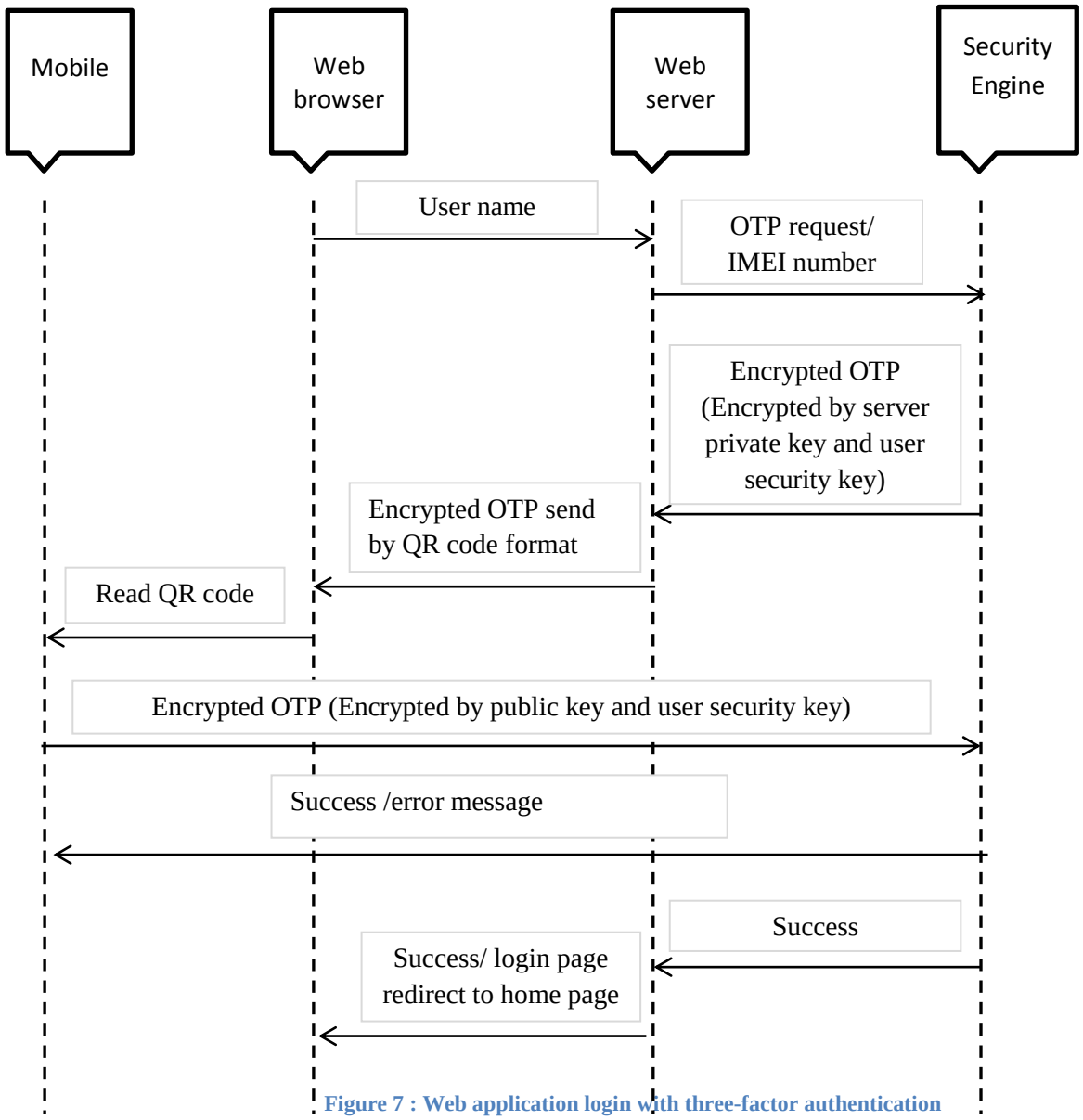


Figure 7 : Web application login with three-factor authentication

The user when tries to login to web application, only he needs to call user name only, all the three factor authentication handled by security engine with smart phone.

The web server request the OTP from security engine, security engine sends private key encrypted OTP. The web server shows the encrypted OTP using QR code format. Mobile read the QR code and decrypted the OTP, then send to security engine, security engine confirm and response to web browser and mobile phone.

3.4.5 Password change

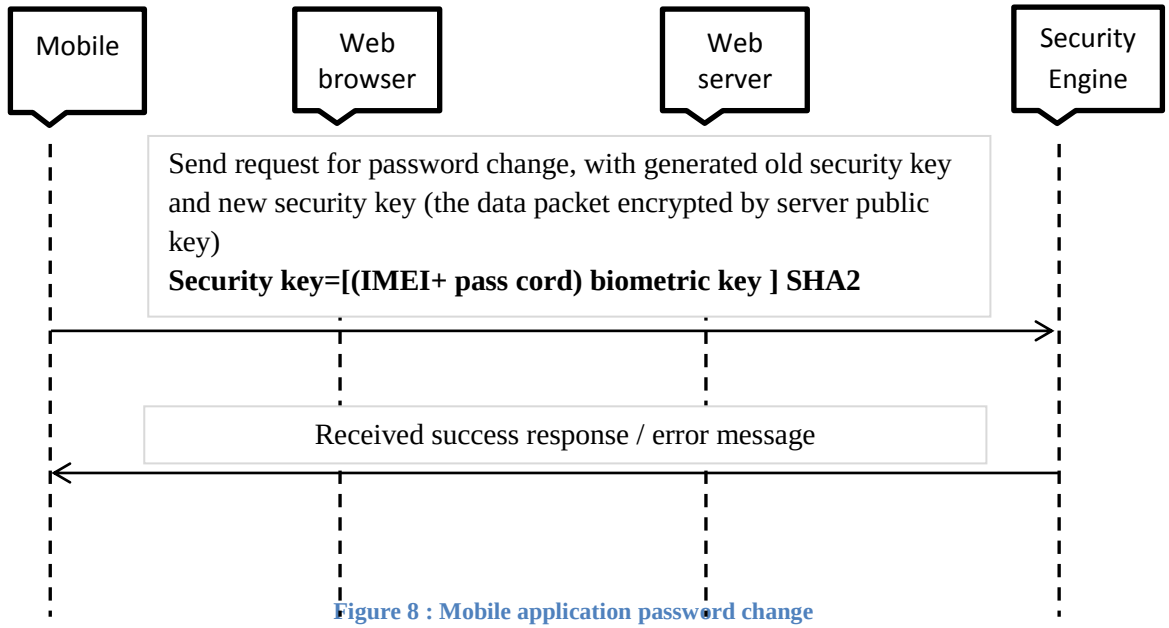


Figure 8 : Mobile application password change

3.4.6 Biometric change

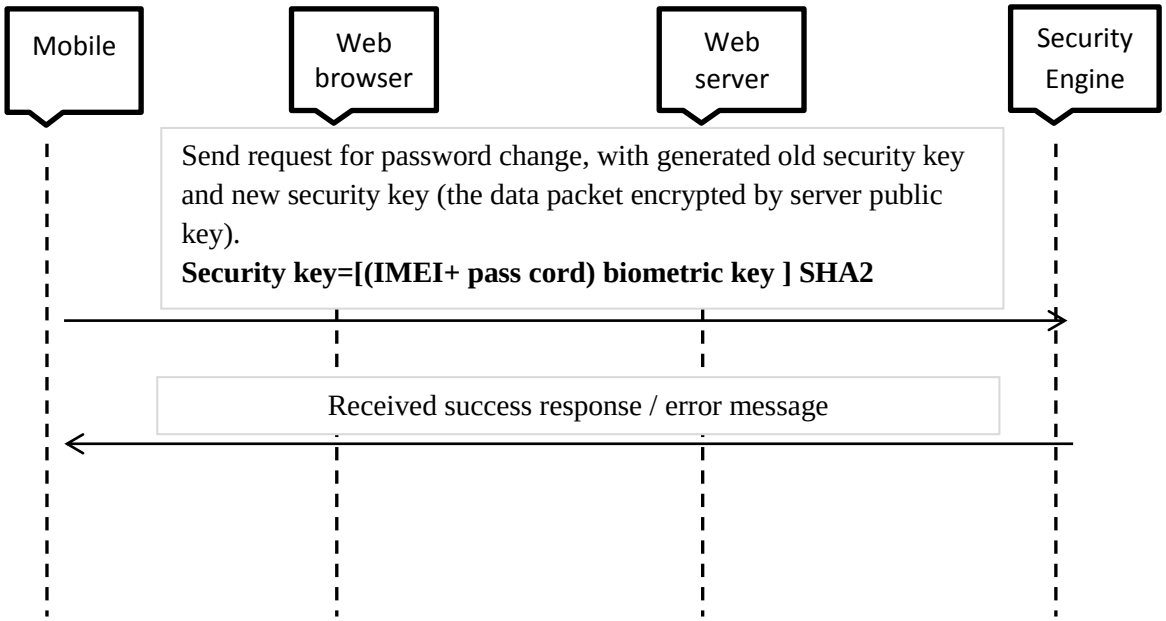


Figure 9 : Mobile application biometric change

3.4.5 And 3.4.6 its show when user has an option to change his password and biometric. When user needs to change the device, then user needs to registration with security engine.

3.5 Functional Overview with flow chart

3.5.1 Initiation download application

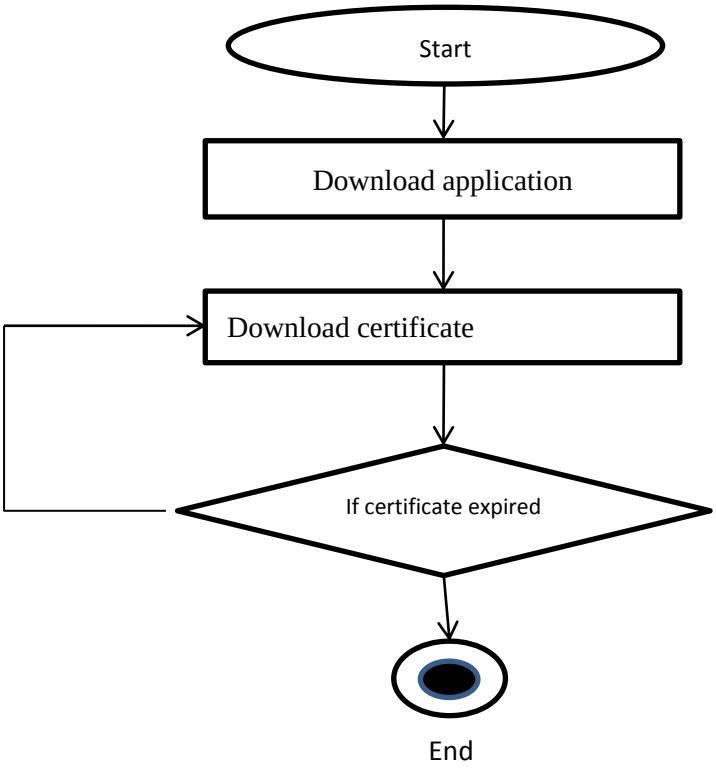


Figure 10 : Flow chart - mobile application downloads

The first step is downloading the mobile application. When download the application the server certificate embed in to the application. In case of revoke or expire server certificate, user has an option to download latest server certificate.

3.5.2 Mobile registration

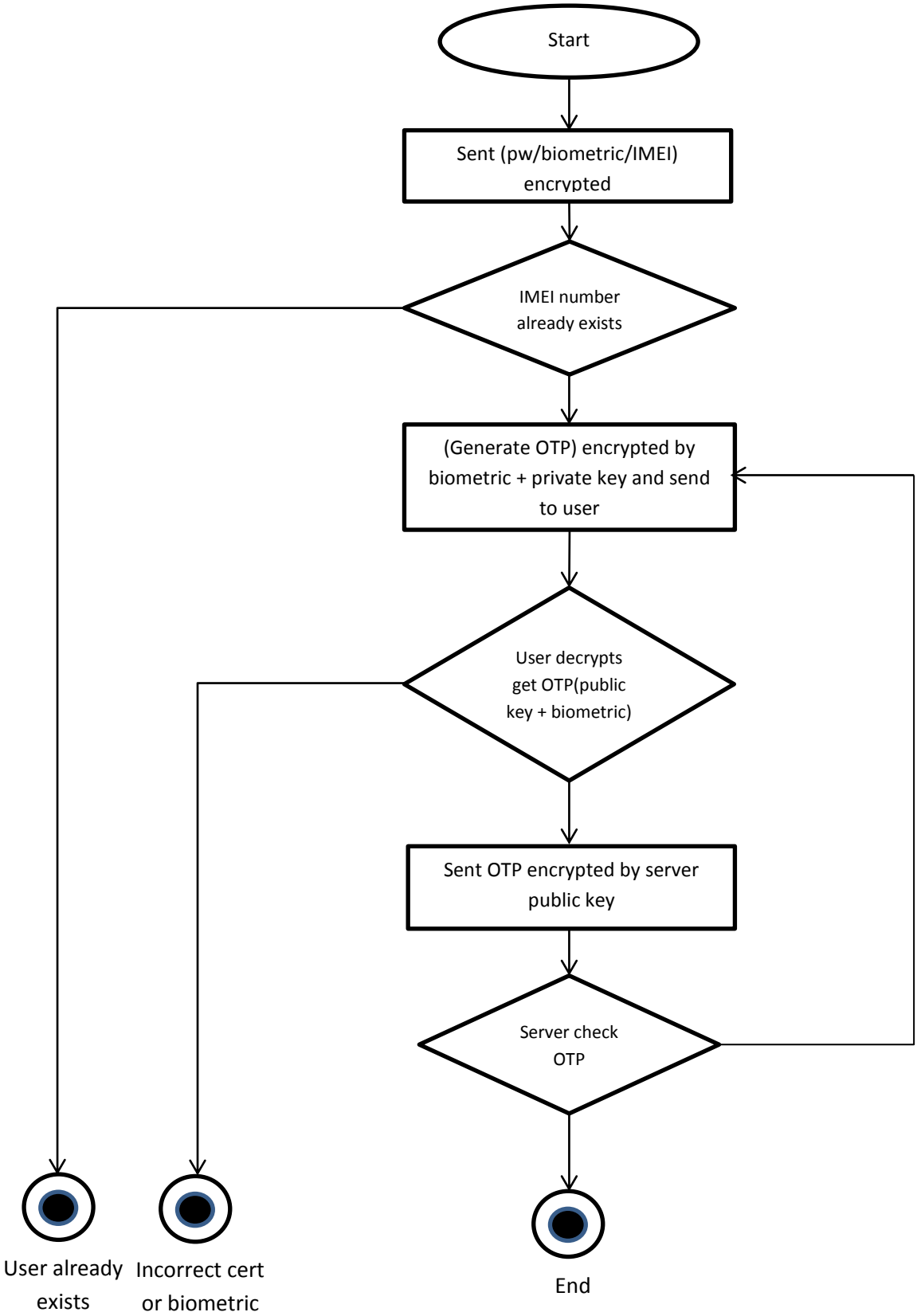


Figure 11 : Flow chart - mobile register with security engine

3.5.3 Add user from web

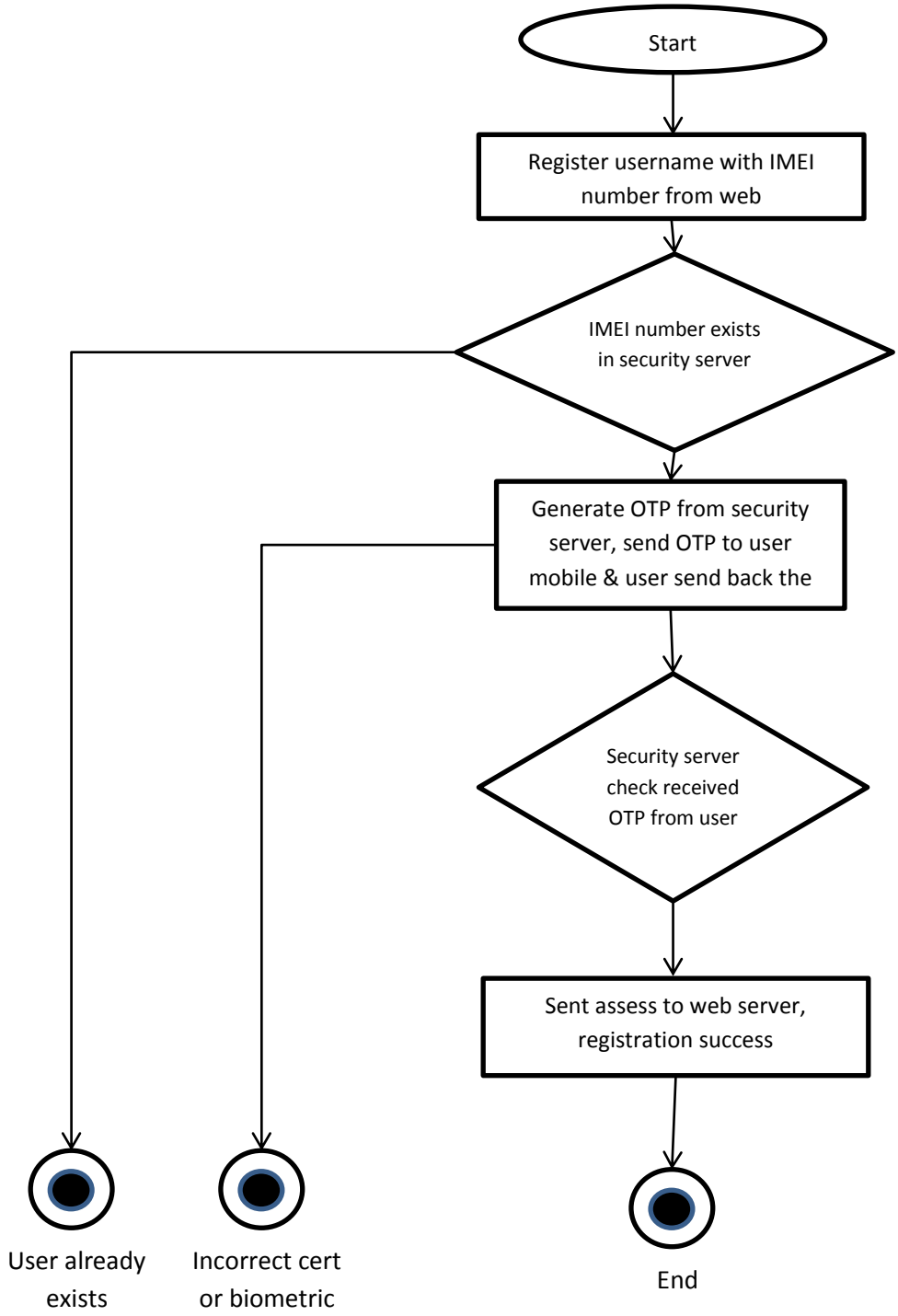


Figure 12 : Flow chart – web application user add

This flow chart showing any web application when add the system user there are two authentication type option user can select, one is user authorize by system level password, other type is user authorize by three factor using third party security engine. Security engine works like single sign on (SSO).

3.5.4 Login Web application

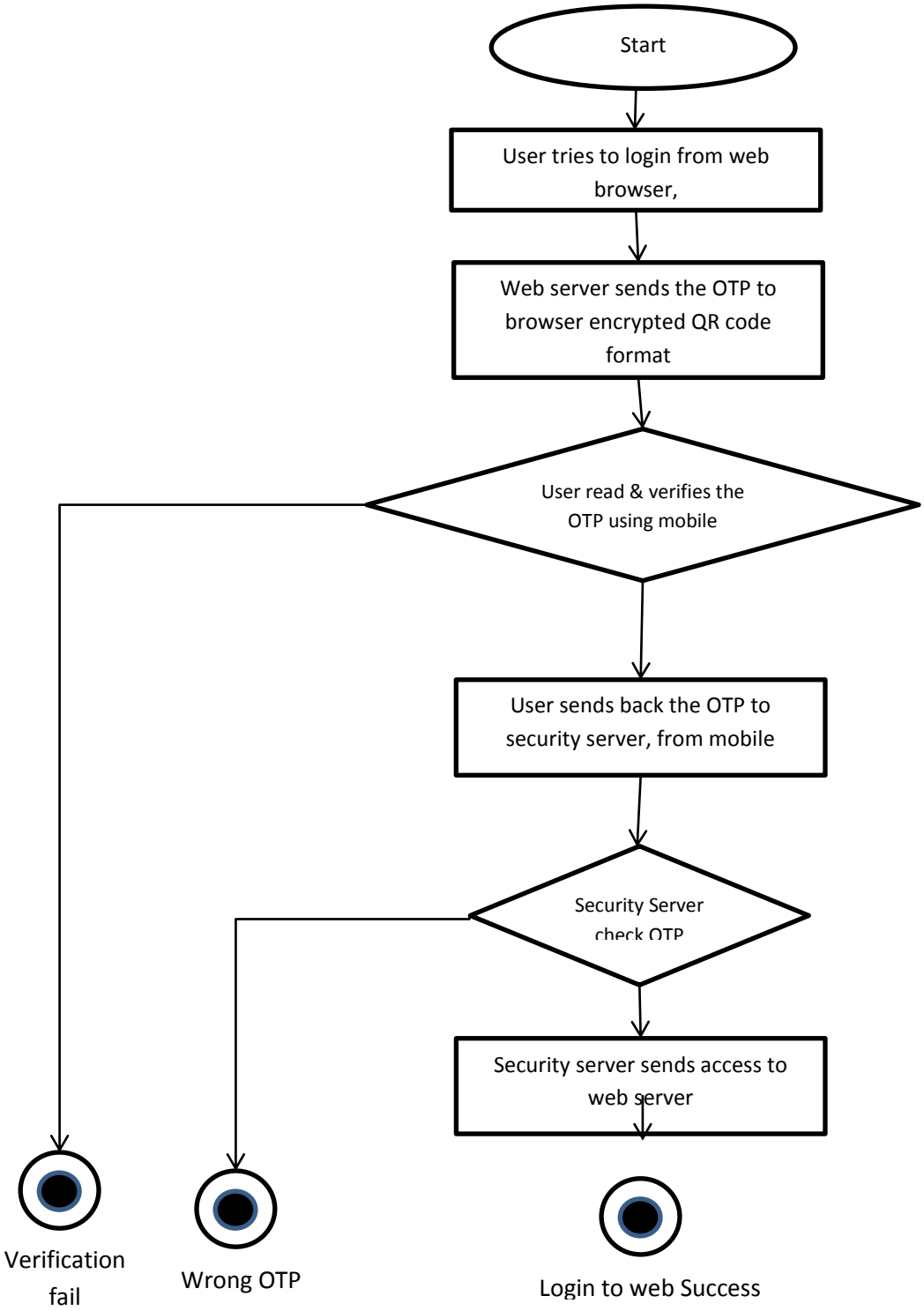


Figure 13 : Flow chart – web application login with three-factor authentication

In the login web application above the flow chart showing web browser get encrypted OTP from its show in browser QR cord format, mobile read QR cord and decrypt using security key retrieve OTP. Then OTP send to security engine to authentication. Web browser continuously checks authentication status on security engine. If authentication status pass browser automatically redirect to home page.

3.5.5 Password change

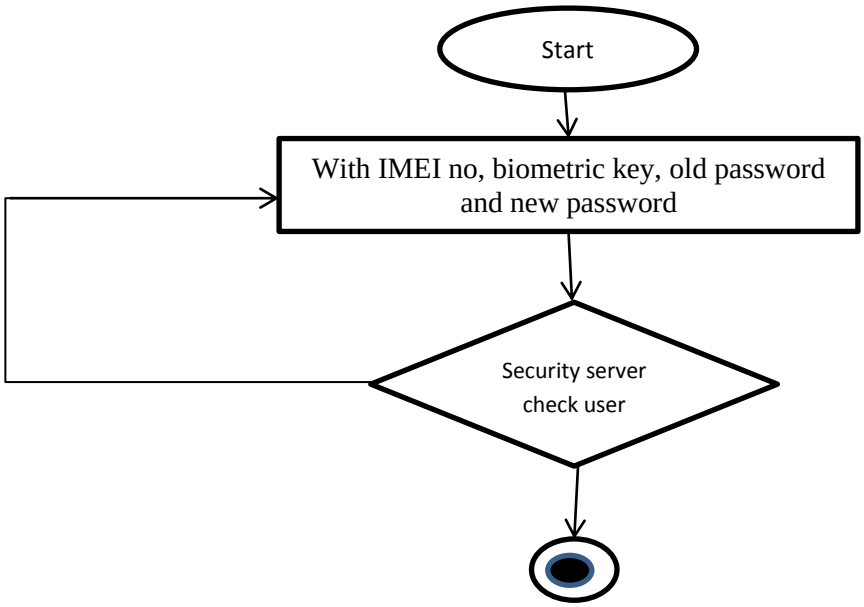


Figure 14 : Flow chart – mobile application password change

3.5.6 Biometric Change

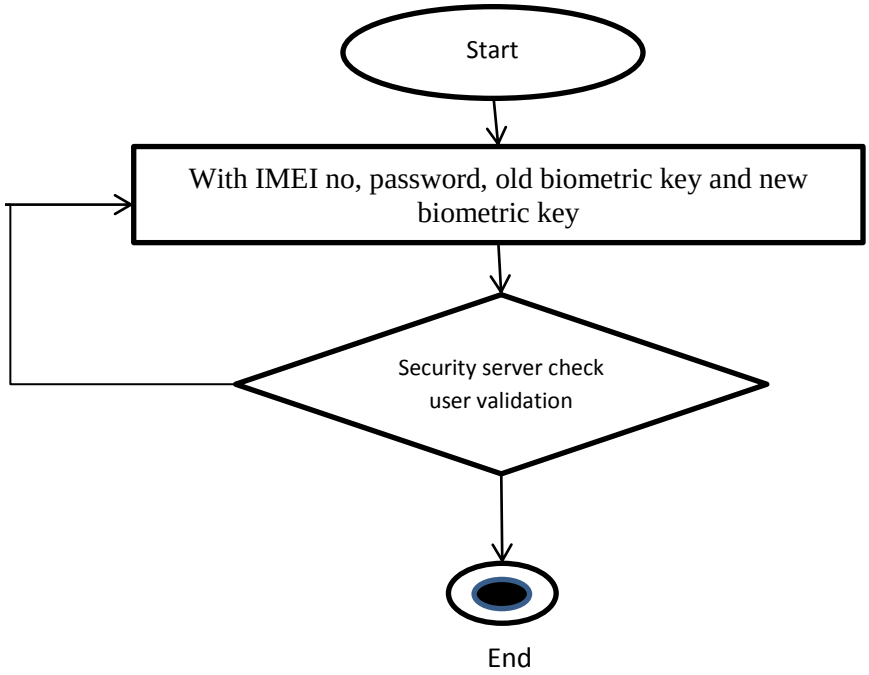


Figure 15 : Flow chart - mobile application biometric change

Above diagram showing pass cord and biometric changing flow chart diagram, both follow same flow both generate security key using (imei, pass cord and biometric) and update to security engine.

3.5.7 IMEI number change (device change)/ device lost

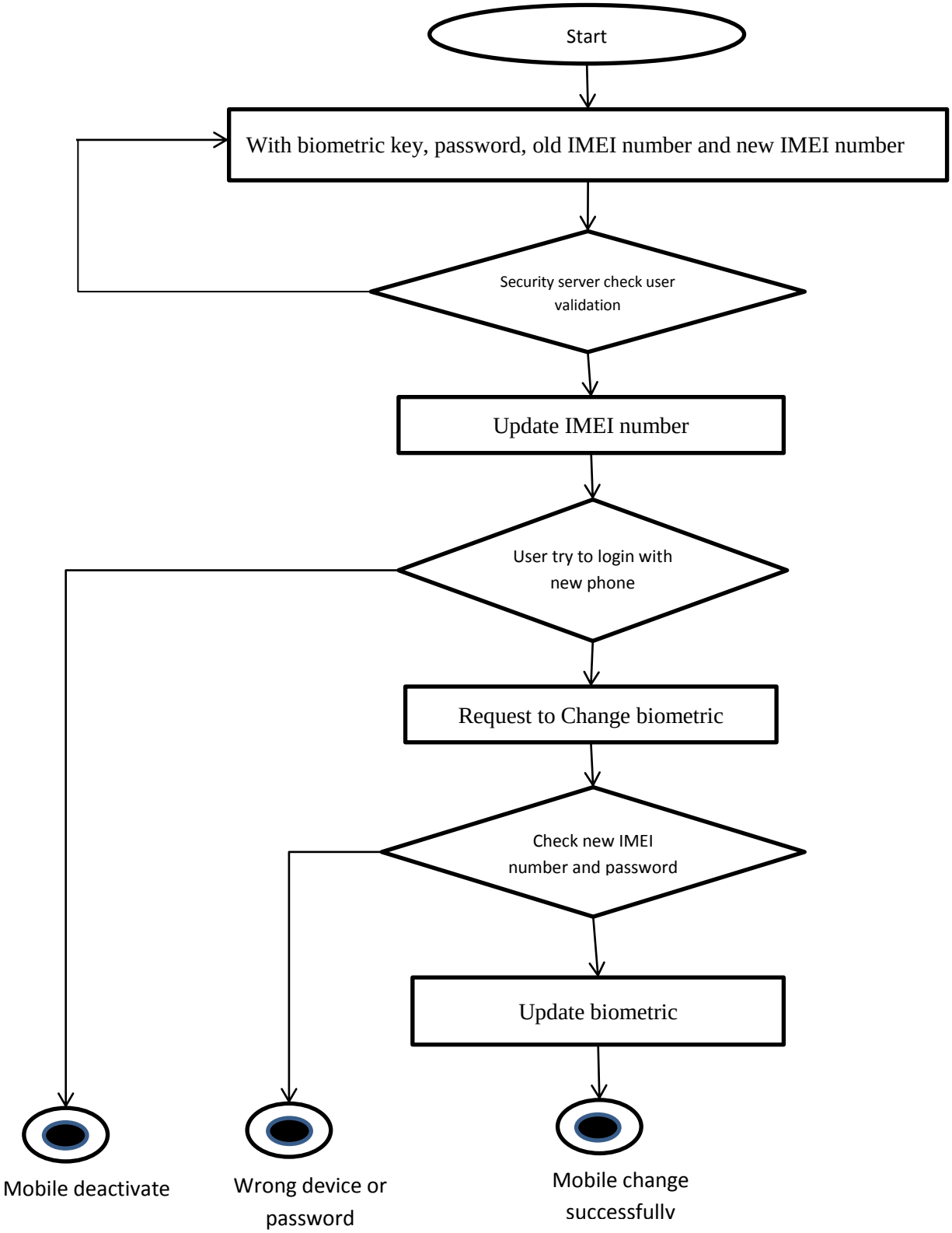


Figure 16 : Flow chart – device change or lost phone

3.6 Methodology

The security server and smart phone communication channel, and how its secure data with five importance concerns. Also the mobile devices provide security to a certain extent than normal communication security data. General cryptography concepts can be used to accomplish the trust between smart phones with security engine.

- **Authentication:** Authentication is the process of proving user identification. Security engine make confirmation smart phone user.
- **Integrity:** Assuring the receiver that the received message has not been altered in any way from the original message.
- **Availability:** System Availability is whether (or how often) a system is available for use by its intended users. This is an integral component of security.
- **Confidentiality:** Ensuring that no one else can read the message except the intended receiver.
- **Non-repudiation:** A mechanism that ensures to avoid that the counter- party later on rolls back the transaction.

4 Implementation

4.1 Main Modules

In the mobile application generate **security key** using IMEI number, pass core and biometric. The security key encrypted, by server certificate. Server certificate downloaded from security server and stored in the mobile application. The security server decrypts the message by server private key and retrieves **security key**. And store local security server databases, such as IMEI number and **security key**. PKI private key stored in the secure server and public key (certificate) stored in all mobile device. IMEI number and pass cord encrypted by biometric key and generate a security key.

On web application connected to the security server, when user registration on web application has two options such as user validate by the local database or user validate by security server using a smartphone. End user login to the web system using three-factor authentication browser show encrypted OTP in QR cord format. Mobile application supported read QR cord and authorization directly send mobile to the security server. The web browser automatically gets access and login to the system. No need to type OTP by the user.

4.1.1 Mobile application

Mobile application security application facilitated on a global range of GPRS, 3G or 4G LTE supported mobile phone brands. In mobile generate unique secure key. Using certificate to encrypted/decrypt the message. On the demo application mobile supported functions are below.

- Smartphone registration
- Added authorize web application
- Login authorizes by QR
- Pass cord change
- IMEI number change(change device)
- Biometric change

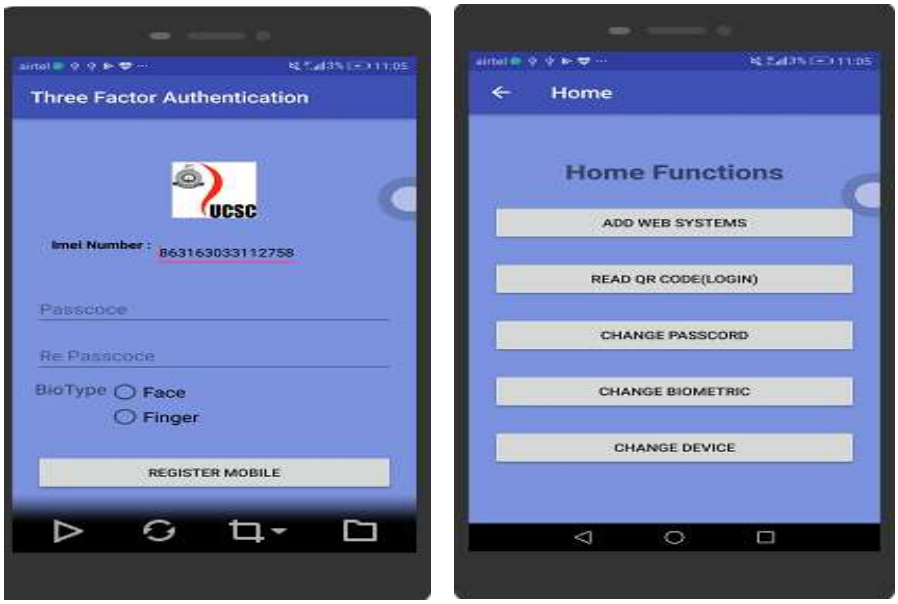


Figure 17 : Mobile application functions list page

Figure 18: Mobile application registration page

4.1.2 Security Engine

Security engine run on glassfish application server, and using restful technology. It will secure in de military zone (DMZ). For the security consent private key stored in keystone using password. On the demo Security Engine supported functions are given below.

- Mobile registration
- Acknowledgment for registration
- Register web application
- Added web application user with a smartphone
- Generated OTP send to the web user
- Validate OTP and update security key (with IMEI, pass cord or biometric).

Security engine stored private key on java key store. And generate CSR file and send that CSR file to root Certificate authentication server to sign. And generate certificate, certificate was published to the entire smart phone device. In case certificate was expired or private key compromised smartphone have option to download the latest certificate from security engine.

If any web application need to interact with security engine. System developer needs to register with security engine and add security engine service to each application. And user needs to authenticate himself using three-factor authentication. User need to register with security engine with smart phone.

Finally user can register with web application, but can select authentication type was inside of the web application or authenticate by security engine. When selecting security engine, engine store application id against user IMEI number with user security key.

4.1.3 Web application

Any web application that was register with security engine. The web server hosted in outside of DMZ. Its contain struts2 web application, mysql database and https/http connectivity. When user register with three-factor authentication sends to security server to register when user login with the three-factor authentication OTP show QR cord format in a web browser. Web browser checks authentication status, with security server time to time. When authentication status changes web browser automatically change from QR cord image to home page. On the demo web application supported functions are given below.

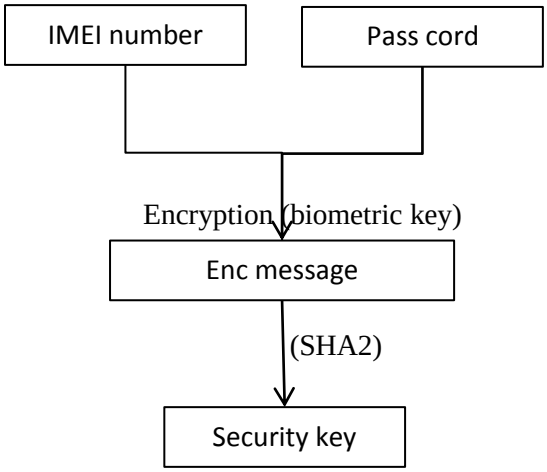
- Register web application with security engine
- When adding the user with three-factor, add IMEI number and application ID to security engine.
- Get OTP(encrypted by security key) from security engine
- Show OTP on QR cord format in browser and continually check authentication status from security engine
- After two mints reset OTP or when authentication status success redirect to home page

There is an option to secure web application more and more using SSL/https use to encrypt login data, SSL encrypt data between web browser and web server, but web server to security engine encrypt data by security key, and server private key and certificate.

4.2 Key Generation

4.2.1 Security Key Generation

Security key generate from smartphone, send to security engine, then security engine store the security key with IMEI number. One way Hash function use as key generator process but SHA-1 has some vulnerable issues. Therefore SHA-2 should be chosen as the Hash function. It has two different block sizes, known as SHA-256 and SHA-512. SHA-256 was chosen as the Hash function because the time it takes to do hashing is lesser compared to the other one.



4.2.2 OTP Generation

Use java random number generator. One time password is valid only for one session or transaction. It's used to standard static passwords, as it eliminates any chance of attacks based on simple knowledge of the password. OTP is difficult for calculation or unable to memorize. It is mostly used in the two-factor authentication process, such as SMS based second-factor validation. It use random number and SHA2, then generate a number. On that long number, last few digits are taken for the OTP.

4.3 Technology background

The three modules such as smartphone, security engine and web server. Below table describes supported minimum software version and server details which required for Implementation. This Software Requirement Specification is prepared by closely analyzing the minimum server requirement to achieve our target.

Index	Requirement	Technology	Minimum supported version
1	Operating System	Linux (Fedora/windows)	Windows 7
2	Application Server	Glassfish\ Apache Tomcat	glassfish 4
3	Database	MySQL	MySQL version 5.1
4	Security engine	Rest full	JDK 8
6	Web Application	J2EE (struts 2)	IE 8 / Firefox/ Chrome
7	Mobile Application	Android	API Level 13

On research the hardware device uses the following such as smartphone, web server, security server and database server. For the demonstration need mobile phone, web application, database server and security engine log files. Android fingerprint application support only grater then API Lever 6. Addition security layer can use SSL and DMZ.

4.4 User Manual

4.4.1 Web user manual

1) Register New User

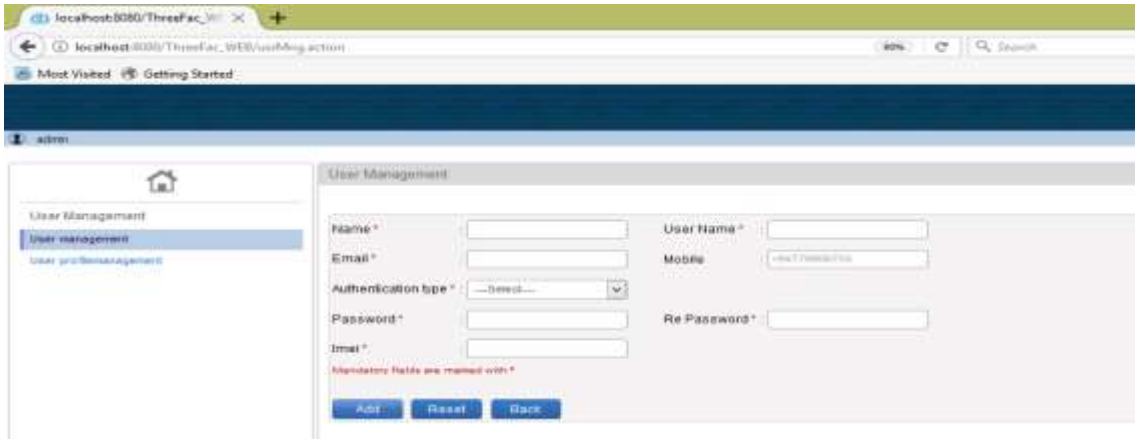


Figure 19 : Web application user registration page

The user registration page is given above, on this page user register with the security server, after click adds button web server send request message to security engine and get response from security engine.

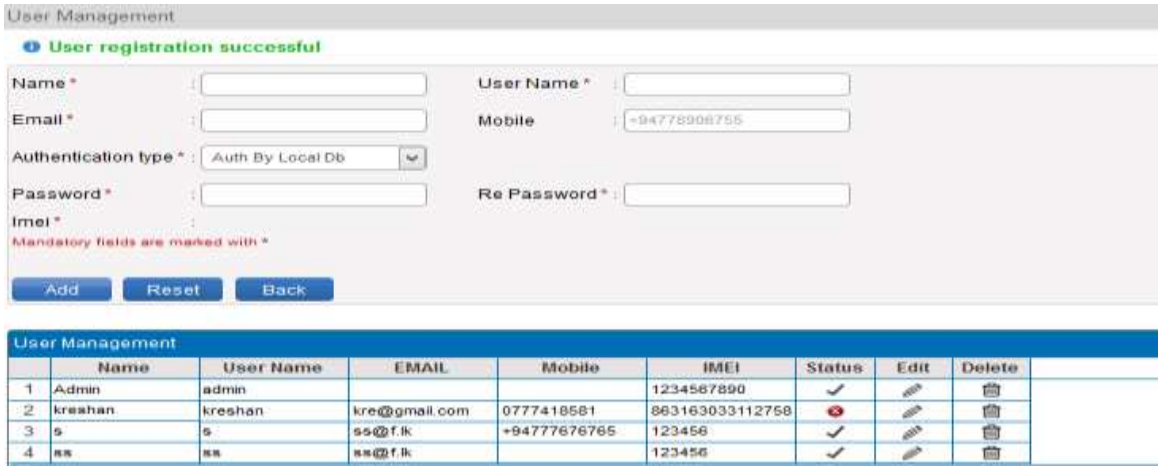


Figure 20 : Web application user registration success page

The page gives success response from security engine, now web user successfully registers with security engine. When register user in web application, there were two options such as authentication by local database detail, or authentication by security engine. When select authentication by local database web server checks authentication itself. When select authentication by security server its send the IMEI number to security engine to register.

Then security engine sends OTP to smartphone which was encrypted by the private key. User decrypts the message using certificate, on smartphone and gets OTP. then user encrypted by his IMEI number, pass cord, biometric key and local server certificate, then send encrypt data to security engine. Security engine stored encrypted user credential and send an acknowledgement to smartphone. And make user register status pending to active status.

2) User Login

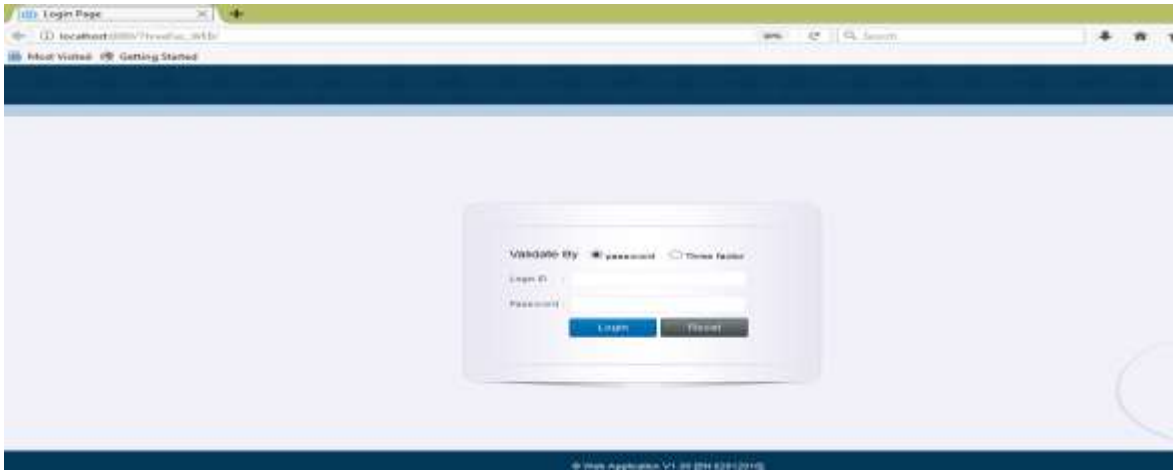


Figure 21 : Web application login page

On the System login page, the user has two options that were validating by password or validate by three factors. When select three factors get registered IMEI number for the user and send to security engine , then security engine generate OTP end encrypt by user security key (40 byte) and private key(1024 bytes) send message (1024 bytes) to web server ,web server decrypt by certificate then send to browser in QR-cord format.

Figure 22 : Web application QR core read page

When user tries to login with three-factor authentication. The encrypted OTP show on the browser in QR-cord format. Then browser time to time checks authentication status with security engine. When authentications status passes automatically web server redirects to home page.

At the same time, mobile read QR-cord and decrypted using security key and retrieve OTP. Then decrypt OTP by the private key and send back to security engine, the engine updates the user login status as pass.

Figure 23 : Web application home page

After authentication status success, web server redirects to home page.

4.4.2 Smartphone user manual

User read all three factors on his smartphone. The authentication flow in a smartphone is following.

1) Registration

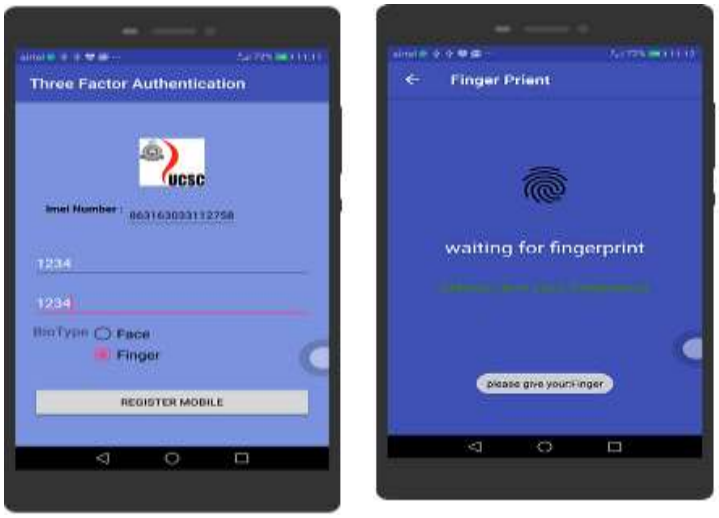


Figure 24 : Mobile application registration page

The device automatically read IMEI number itself, and the user enters pass cord and biometric after that mobile application generates security key using that three-factors. Again encrypt the security key using server certificate. Then send to the security engine, security engine store the security key and send response to smartphone.

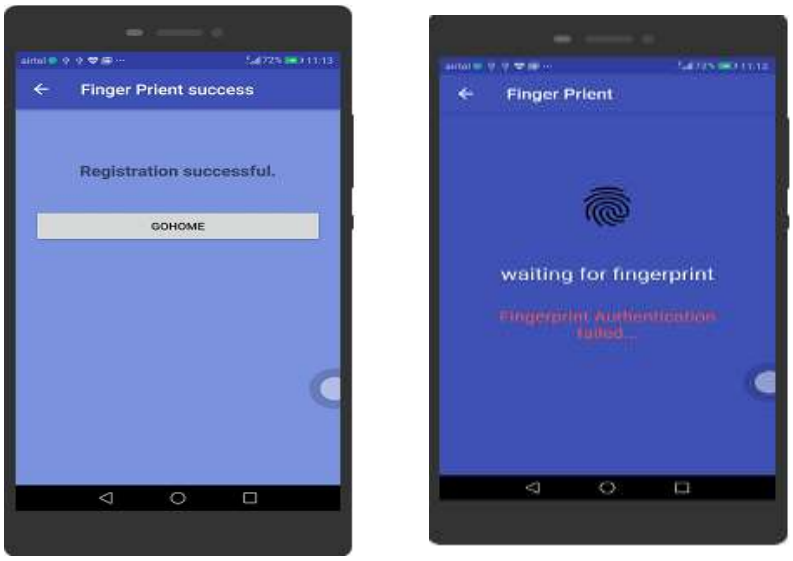


Figure 25 : Mobile application registration success or fail response page

Device registration response from security engine, when the error occurred on the server side or smartphone show error response as registration fails. After mobile registration successful, goes to home page.

1) Homepage

Figure 26 : Mobile application home page

When registration is success it goes to the homepage, it support the number of functions such as add web system id to the security server, read QR-cord read, change pass cord, change biometric and change device.

2) QR-cord read for login authentication

Figure 27 : Mobile application QR read page

Read QR-cord on mobile application from web browser, and then get the encrypted message, after that mobile decrypted using security key and get OTP. Smartphone encrypts

the OTP using server certificate, the encrypted OTP send to the security server for authentication. The server sends an acknowledgment to smartphone and allow QR-code page to home page.

5 Evaluation and Testing

The main aspect of proposed protocol is introducing best three factor authentication protocol with smartphone. Following section will evaluate how the authentication flow has been implemented in the proposed solution. On the evaluation it needs to consider about how the security level, compatible to scales the protocol, how the user interesting, project cost, advantage and disadvantages. In this chapter the proposed protocol evaluated areas as follows.

- Security with scalable
- User Interest
- Project Cost
- Solutions' applicability to the flow of the protocol (Use cases)
- Advantage
- Disadvantages

5.1 Security with scalable authentication protocol

Security of the authentication method is the most critical factor .Authentication purpose using second factor is not protecting user data perfectly; The Second factor not complex process, it was useless implementing at any cost. Later it can be scale three factor authentication or multi factor authentication.

So need to move in to three-factor authentication. Evaluating the authentication protocol is not an easy task. It considers not just the threat prevention aspect of the solution, but also the threat alerting features that just stopping an attack. Now a days smartphone is important in every one’s life. So suing simple mobile application get all three factors of the user, and validate user by single location. It can applicability many area such as remote access platforms, customer websites and point of sale applications, and payment systems.

5.2 User Interest

User Interest is a key factor in the success of the three-factor implementation. Need to be implement user friendly manner that the reason user interest to register with all the application. Users don’t like to carry an extra device to authenticate him, so they end up doing things with their

smart phone. Users want the freedom to login any time and any location, including office, while traveling and any country.

5.3 Project Cost

Total Cost of the solution can be calculate by variance in implementation, deployment, support costs and maintainers cost.to calculate the cost of the project need to consider thinks such as,

- Hardware and Software cost – perches servers and application servers
- Development and Deployment - Developer cost , end user training cost, marketing cost
- User Support Team – need to online support team to register application with security engine
- Certificates cost – perches PKI certificate by CA root server

5.4 Solutions' applicability to the flow of the protocol (Use cases)

On the three-factor authentication protocol to evaluate, here using some of use cases, the user case have use case id, and satisfy some pre condition to test that use case. Below chart show some use cases such as smart phone registration, user add to web system, web system register with security engine, user login to web system, user pass cord change, user biometric change and device change.

Below chart basic flow following success path of the use case. Alternative flow is showing fail case of the use case, alternative flow id is showing the error case from which level from successful path.

5.4.1 Register smartphone

Scenario	A non-existing customer attempts to register to the Security Engine.
Use Case ID	UseCase-01
Purpose	This will enable the non-existing user is registered with security Engine.
Actor	End-user (with smartphone), Security Engine, Security engine database, SSL cert, android play store
Pre-Conditions	
1. The smartphone should be connected to the Internet. 2. User fingerprint locally registers with the smartphone.	
Basic Flows	
1. Customer downloads the application from App Store/Play Store. 2. The customer selects the option 'Registration'. 3. A customer fills the following form. • Pass cord • IMEI number(automatically loaded) • select the biometric type • load biometric 4. The customer selects the button 'Register. 5. Smartphone sends encrypt key & IMEI number to security Engine, with fully encrypted by the certificate. 6. Security Engine generates OTP and sends OTP to mobile encrypted by the private key and smartphone key. 7. Smartphone read the OTP and send back to security Engine. 8. Security Engine validates OTP and sends success response to mobile, and phone details stored in security engine database. 9. Mobile displays a success message to the customer.	
Alternate Flows	
3. Customer gave the Wrong fingerprint 5. certificate wrong cannot decrypt the message by security Engine 7. When certificate expire or mobile user credential is wrong cannot read OTP. 8. The OTP will be expired after 2 minutes on security engine server 9. Registration fails, with the wrong OTB or expired OTP.	
Output	
1. Registration success, go home page **2.** Mobile redirect to again registration page with error message	
Remarks	
1. When first-time registration success. then next time registration page not appear **2.** The same device cannot register to security Engine two times	
References	
1. Flowchart diagram reference – figure 11	

In this flow chart when user download the application, the server certificate also include with that. When certificate was expired user has an update option to download new certificate. When security engine get mobile detail send OTP to mobile to acknowledgement. Mobile decrypted generated security key get OTP and send back to security engine.

5.4.2 User add to web system

Scenario	Add user to web application, with authentication along with security server
Use Case ID	UseCase-02
Purpose	The new user login to web application with authenticating by smartphone, without using password or using password
Actor	End-user (with smartphone), web application user, Web server, Security Engine, Web server database, Security Engine database

Pre-Conditions

1. The smartphone should be connected to the Internet.
2. web application server has a connection with security Engine

Basic Flows

1. click ‘user add’ page in a web application
2. The user adds page keys-fields as followings.
   - Username
   - mobile number
   - Email
   - Select authentication type (Password/IMEI number)
   - Enter password/ IMEI number
3. The customer selects the button ‘ADD.
4. If authentication type was ‘IMEI number’ call security Engine to add system user
5. Web user added.
6. security engine send alert to a smartphone using Google firebase service
7. User Smartphone send request to security engine, to authorize user which was added by web application
8. Security engine generate OTP for (IMEI+ web system) and send OTP to smartphone
9. Smartphone read the OTP and send back to security engine
10. security engine validate OTP and update the status against(IMEI + web system) , and send a success message to smartphone
11. Web user added successfully, end user now can authorize himself by smartphone.

Alternate Flows

4. When authentication type was ‘password’, web application locally store password and not sending a request to security Engine. and if Security Engine doesn’t register IMEI number send an error message to the web
9. without giving authorize by smartphone, the user cannot login to the web system
10. (IMEI + web system) OTP will expire within 24 hours

Output

1. The user adds to web system successfully.
2. Smartphone get authentication success message from security engine

Remark

When new users add to web system, the user has two options such as password validate by local database, or authentication validates by the three-factor security server.

References

1. Flow chart diagram reference – figure 12

User add use case show user add in web page. User ads have two options such as password validated locally or password validate by security engine. When select security engine, security engine check given IMEI number already registered send success message, it is the success path of user add.

5.4.3 User login to web system

Scenario	User login to web system, with authorizing by smartphone
Use Case ID	UseCase-03
Purpose	User login to web system, with authorizing by smartphone or password
Actor	End User(smartphone), Web application, Security Engine, web application database, Security Engine database

Pre-Conditions

1. The device should be connected to security engine.
2. Web server connects to security engine.
3. Smartphone register with security engine.

Basic Flows

1. User login to web application
   - Username
   - Authentication type(local database password/ three-factor server)
   - Password / IMEI number
2. Select authentication type was local database password or three-factor server
3. If select local database password, validate on web application database
4. If select three-factor server authentication
5. The user selects the button ‘Login’.
6. The web server sends IMEI number and server id to security engine.
7. Security engine validate IMEI number with server id
8. Security engine generate OTP encrypted by user key and send response to web server
9. Web server show encrypted message in QR-code format
10. Smartphone read QR cord and decrypt the OTP using user key, send back OTP to security engine
11. Security engine validate OTP, give access to web application
12. User web application automatically login to the system.

Alternate Flows

2 When authentication type is ‘local database password’, validate user password locally
7 if wrong IMEI number send an error message to web application server
10 when user key wrong, show error message in decryption.
11 when OTP is wrong or expired send an error message to mobile.

Output

1. Login credential are correct in local database login to home page
2. Login credential are correct in three-factor server to show QR cord/ error message
3. in smartphone when reading QR core show authentication success or authentication error

Remark

- user key= sha2(encrypt by finger print key (passcode+ IMEI))

References

1. Flow chart diagram reference – figure 13

When user try to login to web system user type user name and password or IMEI number, depend on his selection in the password field user decide to type password or IMEI number. When select IMEI number, web server checking authentication in security engine.

5.4.4 User pass cord/user biometric change

Scenario	User pass cord or user biometric change
Use Case ID	UseCase-04
Purpose	User pass core or user biometric change, updated user key
Actor	End User(smartphone), Security Engine, Security Engine database, Certificate
Pre-Conditions	
1. The device should be connected to security engine. 2. devise register with security engine	
Basic Flows	
1. User login to smartphone three-factor application 2. In Homepage, selects ‘password change’ or ‘biometric change’ button.(pass cord) 3. get the biometric key, old pass cord, and new pass cord 4. select button ‘change pass core’ 5. old user key and new user key send to the security engine 6. the response comes from the security engine 7. pass cord change successfully	
Alternate Flows	
3 wrong passcode 7 pass cord change fail, old pass cord wrong	
Output	
1. password change success full/fail	
Remark	
• old user key= sha2(encrypt by finger print key (passcode old+ IMEI)) • new user key= sha2(encrypt by finger print key (passcode new+ IMEI))	
References	
1. Flowchart diagram reference – figure 14 & figure 15	

Pass cord change and biometric change follow same use cases. when biometric or pass cord change in mobile locally generate security key , then mobile send ole security key , new security key and IMEI number to security engine. Security engine update security key, after that only new credentials work for authentications. Security engine check is it register system and register user, then generate OTP and stored in security engine and send OTP to web server

5.4.5 User device change

Scenario	User need to change new smartphone
Use Case ID	UseCase-05
Purpose	User needs to change three-factor authentication smartphone to new smartphone
Actor	End User(smartphone), Security Engine, Security Engine database, Certificate
Pre-Conditions	
1. The device should be connected to security engine. **2.** devise register with security engine	
Basic Flows	
1. User login to smartphone three-factor application 2. In Homepage, selects ‘change device’ button. 3. get old IMEI and new IMEI. old IMEI number automatically loaded 4. select button ‘change device’ 5. generate old user key and new user key. 6. old user key and new user key send to the security engine 7. the response comes from the security engine **8.** device change successfully **9.** can log in to the new device, the old device was deactivated	
Alternate Flows	
7 device change fail, old pass cord wrong	
Output	
2. password change success full/fail	
Remark	
• old user key= sha2(encrypt by finger print key (passcode old+ IMEI)) • new user key= sha2(encrypt by finger print key (passcode new+ IMEI))	
References	
3. Flowchart diagram reference – figure 16	

User device change is most important part in the protocol, when user lost device no option to retrieve data. When user needs to change the device, using old device update to new IMEI number then old device will be deactivated. After that only new device can do authentication process.

5.5 Advantage

Now a day’s system hackers are grown well, so it needs to update security protocol, in these thesis using three factors to authenticate to higher the secure layer. Other system not stored user password information, only security engine store secure information.

5.6 Disadvantage

Authentication check in single point, security engine checks the authentication all the times. If authentication server down all the application strut with issue. Security engine needs high availability (HA) to reuse transaction failure.

6 Conclusion and Future Work

6.1 Conclusion

The model proposed in this paper three-factor authentication using a smartphone by all registered web system. For using this authentication protocol any of the web system and any user need to be registered with security Engine. The three-factor register's user can authenticate by using his smartphone with three factors such as IMEI number, pass code and biometric kind of authentication offers better security and privacy. In this security protocol any system when need to authorize by three factor, we can do authentication part is single point using security server. Three-factor authentication approach is used for the authentication and authorization of the user, which increase the confidentiality and integrity of the user credential. PKI infrastructure used in security engine, security server save private key in a secure location and send the certificate (public key) to all the smartphones. The smartphone and the security engine which guarantees the isolation and safe execution of the security protocol. Use Google time synchronizer in smartphone and security engine to avoid a replay attack. Technologically reasonable protocol to achieve security communication in a responsible manner, complying with security engine goals; confidentiality, integrity, availability, authenticity and non-repudiation of communication message was proposed.

6.2 Future work

The limitation was when user lost his smartphone; the user has no option to change the device, only option he needs to register the system again. Biometric key only store in the mobile device there is a drawback, so need to find the best way to store biometric in security Engine. In the feature, it can support all the type of biometric. There is an option sms based or email based validate user and rollback lost phone details.

REFERENCES

The following documents and websites have been referred to while writing this report.

[1] R.Ghițescu and A.Carpenter, “Universal 2nd Factor Software Token,” 2016.

[2] S.Vaithyasubramanian, A.Christy and D.Saravanan, “Two Factor Authentication for Secure Login Support of Effective Information Representation and Network Security,” ARPN Journal of Engineering and Applied Sciences, Vol.10, Issue.5, pp. 1819-6608, 2015.

[3] F.Aloul, S.Zahidi and W.El-Hajj, “Two Factor Authentication Using Mobile Phones”, 2009.

[4] Ashish Shrestha and Almurrani Balsam, “Multi-biometric systems”, 2014.

[5] K.S.Babu and A.F.Saleem, “A Generic Frame Work for Three Factor Authentication Preserving Security and Privacy in Distributed System,” International Journal of Scientific Engineering and Technology Research, Vol.03, Issue.03, pp. 0484-0490, 2014.

[6] M.Alizadeh, W.Haslina Hassan and T.Khodadadi,"Feasibility of Implementing Multi-factor Authentication Schemes in Mobile Cloud Computing," 2014 Fifth International Conference on Intelligent Systems, Modelling and Simulation,Vol.,Issue.,pp. 615-618, 2014.

[7] P.Soni and M.Sahoo,"Multi-factor Authentication Security Framework in Cloud Computing," International Journal of Advanced Research in Computer Science and Software Engineering, Vol.5, Issue.1, pp. 1065-1071, 2015.

[8] Mobile Multi-Factor Authentication, IBM Knowledge Center, available online:https://www.ibm.com/support/knowledgecenter/en/SSPREK_9.0.2/com.ibm.isam.doc/config/concept/con_mmfa.html. last accessed on 28/07/2015.

[9] S.Lakshmi, N.S.Annapurna and T.Latha,"Security Analysis of Factor Authentication Schemes for Banking," ARPN Journal of Engineering and Applied Sciences, Vol.10, Issue.8, pp. 1819-6608, 2015.

[10] S.Yadav, P.Patil, M.Shinde and P.Rane, "Android-Based Mobile Payment System Using 3 Factor Authentication," International Journal of Emerging Technology and Advanced Engineering, Vol.4, Issue.3, pp. 2250-2459, 2014.

[11] I.A.Lami, T.Kuseler, H.Al-Assam and S.Jassim, "Mobile phone based multifactorbiometric authentication with time and location assurance," 18th Telecommunications forum TELFOR, pp. 152-155, 2010.

[12] F. Aloul, S. Zahidi, W. El-Hajj, "Multi Factor Authentication Using Mobile Phones," International Journal of Mathematics and Computer Science, Vol.4, Issue.2, pp. 65-80, 2009.

[13] PCI Security Standards Council, " Multi-Factor Authentication", vol 1.00, 2017. online:https://www.pcisecuritystandards.org/pdfs/Multi-Factor-Authentication-Guidance-v1.pdf. last accessed on 01/11/2017.

[14] M.Roy, Prof. A.Verma," Multi-Factor Authentication Scheme for E-Services in Cloud Computing," International Journal of Innovative Research in Computer and Communication Engineering, Vol.5, Issue.1, pp. 2320-9801, 2017.

[15] B.Chaimaa, E.Najib, R.Hilal,"Authentication Mechanisms in cloud Computing Environment," International Journal on Information Technologies & Security, Vol.3, pp. 63-84, 2017.

[16] N.Gupta and R.Rani,"Implementing High Grade Security in Cloud Application using Multifactor Authentication and Cryptography," International Journal of Web & Semantic Technology (IJWesT), Vol.6, Issue.2, 2015.

[17] M.Khamis, R.Hasholzner, A.Bulling, F.Alt,"Two-factor Authentication on Public Displays Using Gaze-Touch passwords and Personal Mobile Devices," Vol.1 2017.

[18] M.Aldwairi, S.Aldhanhani, "Multi-Factor Authentication System", 2017. online:https://www.researchgate.net/publication/319312344_Multi Factor Authentication System. last accessed on 30/11/2017.

[19] UCLA information technology services, "Deploying Multi-Factor Authentication with UCLA Logon", 2016. online: http://www.csg.ucla.edu/documents/2010/05b-iamucla-mfa-csg-20160426.pdf. last accessed on 05/11/2017.