ST. MARY'S UNIVERSITY

SCHOOL OF GRADUATE STUDIES

STRENGTHENING THE SECURITY OF MOBILE CLOUD COMPUTING USING HYBRID AUTHENTICATION TECHNIQUES

By
Endale Amdie Gebremeskel

Advisor: Asrat Mulatu (PhD)

February, 2021
Addis Ababa, Ethiopia

STRENGTHENING THE SECURITY OF MOBILE CLOUD COMPUTING USING HYBRID AUTHENTICATION TECHNIQUES

By

Endale Amdie Gebremeskel

A thesis Submitted to the Faculty of Informatics, St.Mary's University, in Partial fulfillment of the requirements for the degree of Master of Science in Computer Science.

February, 2021

Addis Ababa, Ethiopia

St. Mary's University
Faculty of Informatics
Department of Computer Science

STRENGTHENING THE SECURITY OF MOBILE CLOUD COMPUTING USING HYBRID AUTHENTICATION TECHNIQUES

By

Endale Amdie Gebremeskel

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

_____

Internal Examiner

_____

External Examiner

_____

Dean, Faculty of Informatics

February, 2021

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all source of materials used for the thesis work have been duly acknowledged.

Endale Amdie

_____

Full Name of Student

_____

Signature

Addis Ababa, Ethiopia

This thesis has been submitted for examination with my approval as advisor.

Asrat Mulatu (PhD)

_____

Full Name of Advisor

_____

Signature

Addis Ababa, Ethiopia

February, 2021

ACKNOWLEDGEMENTS

First and for most I would like to thank the Almighty GOD for his unending helps and blessings to complete my thesis, without his blessings and support I wouldn't have been write a single word. Next, I would like to express my gratitude and thank my Thesis Advisor Dr. Asrat Mulatu for his excellent guidance, patience and support in completing this study.

I would also like to thank my friends and colleagues especially for Abel Dagnew for his consistently giving me the technical and non-technical assistance.

Last but not the least; I'm very grateful to my parents, brother, sisters and friends for their encouragement, inspiration and unequivocal support. Without their encouragement and unconditional support, I would not be where I am today.

List of Acronyms

AES:	Advanced Encryption Standard
API:	Application Programming Interface
BioAaaS:	Biometric authentication as a service
CEM	Common Evaluation Methodology
CC:	Cloud Computing
CSP:	Cloud Service Provider
DTW:	Dynamic time warping (DTW)
IaaS:	Infrastructure as a service
IDC:	International Data Corporation
IMSI:	International Mobile Station Identification number
MCC:	Mobile Cloud Computing
MSP:	Manages Service Model
OLOF:	Orthogonal Line Ordinal Features (OLOF)
PaaS:	Platform-as-a-service
PIN:	Personal Identification Number
QoS:	Quality of Service
SaaS:	Software as a service
SMCBA:	Securing Mobile Cloud using Biometric Authentication

Abstract

The mobile cloud computing has become a popular business transaction platform today because modern mobile sets are not used just for making calls and sending messages. They are increasingly being used in mobile cloud computing (MCC) to store sensitive and critical information as well as to access sensitive data using the Internet via cloud service provider (CSP). The majority of these devices use inherently weak authentication mechanisms, based upon passwords and personal identification numbers (PINs). But it is not secure way for authenticating users and also it is difficult to confirm that the demand is from the rightful owner. Authentication is one of the main security problems in mobile cloud computing. This study focus on strengthening user authentication in the mobile cloud computing and proposes new authentication security architecture as well as develop a new hybrid authentication on the mobile cloud environments.

The design science approach is applied in this study to assess the recent work on the area of data security related to mobile cloud computing, authentication security issues and solutions and reference architectures; and proposed a secured hybrid authentication technique for the mobile cloud computing environment. The evaluation indicates that the proposed strengthening security of mobile cloud computing using a hybrid authentication technique is more secure. The evaluation result shows that using a combination of username/password and fingerprint authentication is a viable option for strengthening user authentication on the mobile cloud computing environment.

Keywords: Mobile cloud computing, cloud computing, Authentication, biometric, fingerprint and cloud service provider

List of Figures

List of tables

Table of Contents

CHAPTER ONE

INTRODUCTION

1.1 Background

Present days mobiles are not used just for making calls and sending messages. They are increasingly being used in Mobile Cloud Computing (MCC) to store sensitive and significant information as well as to access sensitive data using the Internet via cloud service provider (CSP). Mobile cloud computing is a combination of Cloud Computing (CC) and mobile communications. The use of MCC helps to reduce running cost and expansion of mobile applications. One of the important challenges in mobile cloud computing is security and privacy. MCC is used to provide rich computing resources to mobile networks and differs from mobile computing, which runs cloud-based web applications rather than native applications.

The development in technology has also brought many new security threats within it. Every user wants the high protection of their data and is curious about it. Recently, the cloud computing is becoming a new hot technology. And the security solution for it has become a research focus. With the development of the mobile cloud computing, new security issues are there, which needs more security approaches [1].

From the MCC security and privacy authentication has a big role in security. *Authentication* is the process of comparing the credentials with the registered credentials which are stored in the database and if the matching is successful then user is allowed to proceed. If the matching fails the user is not allowed to proceed to the next steps and is informed to provide valid credentials.

Authentication is the process of recognizing a user's identity. It is the mechanism of associating an incoming request with a set of identifying credentials. The provided credentials are compared with the data in a file in the authorized user information database on the local operating system or on the authentication server. In this study, we focus on how to use the two authentication methods in mobile cloud computing. By using these two authentications which is the combination of username/password with biometric (fingerprint) techniques; we propose new authentications together to strengthen the authentication in the MCC security.

Password authentication: - The use of the password is a frequent form of usage for authentication. This method requires that each user must have a username and a password/PIN. It requires the users' to provide the password with the specific characters. This technique is cheaper and its use is most popular.

Biometric Authentications: - The features used in this approach to authentication can't be stolen, forgotten, or forged. For instance, a user's fingerprint, eyes, face, hand structure, or voice cannot tamper. These are the features that are most helpful for authentication because every human being has his/her characteristic features which cannot be forged or stolen.

Several studies have been done to suggest suitable authentication schemes in cloud computing [6] [10]. However, authentication in the MCC, as one of the most important security measures, has not yet been fully studied because till now there are security threats on the MCC environment [13]. Moreover, several attempts have been made to study various aspects of MCC.

Authentication is a process of verifying the identity of an individual or an object such as a mobile device. It requires the user or object to furnish its credentials which is compared with the ones already present in the database [2] [3]. This is important for providing security and privacy to the users [4], particularly for applications transmitting sensitive and personal data to the cloud. Some of the challenges of authentications include complexity in providing user credentials, number of handshakes required for verification, and delay. Our work can solve these authentication complexities by making fast and secure authentication. In MCC, the task of authentication becomes more complex, as the communication between the cloud and mobile device takes place over various wireless networks (Wi-Fi, 3G, and 4G). Authentication delay is important in MCC for real-time applications such as online movie watching, online game playing [5].

1.2 Motivation

The primary focus of this work is to design and develop a new system by combining username/password with fingerprint authentication methods to authenticate the users correctly. In the new era, strong authentication techniques are required to detect security breaches. We believe that there is a need for practical mechanisms that can be easily deployed to improve

authentication security with a minimal impact on usability without necessarily having to supplant current PIN or password-based authentication that is already familiar to users. The general motivation behind the research in this paper is to contribute towards identifying and developing mechanisms to fulfill the aforementioned need.

And also some of the cloud computing service providers do not have strong security aspects and still use *insecure or weak authentication methods,* enhancing cloud computing security is still an open research area; as hackers continuously gaining more knowledge and experience [60]. Precisely, clouds now are used by smartphones and multi Internet access devices and will be used as universal technology in the world requiring security policies with a strong authentication and access control process to protect the clouded data and ensuring availability and privacy. This triggers the contribution of this paper: proposing a security framework to be used in personal mobile smartphones (e.g. Android, OS …), portable devices, computers, and workstations to access the cloud storage service securely and secure privacy and protection of clients' sensitive and personal data.

1.3 Statement of the Problem

Numerous challenges facing Mobile Cloud computing (MCC) include trust, security, and privacy. For MCC to become an effective computing paradigm, effective solutions have to be found for these issues, especially for security. One of the important aspects of security is authentication. Any approach to security must include effective techniques for authentication taking into consideration the limited resource environment present in a mobile device.

The migration of private and enterprise data to the cloud raises security and privacy issues. To access these sensitive data only by the legitimate users, an authentication protocol is used. Traditionally, a user provides his or her password to the requested server for authentication, which may be attacked. In mobile cloud, legal user authentication becomes an important issue.

Mobile devices have found an important place in modern society, with hundreds of millions currently in use. The majority of these devices uses inherently weak authentication methods; based upon *passwords* and *personal identification numbers (PINs).* The main problem with passwords or PINs is that they may be forgotten, stolen, spoofed, or guessed by an attacker.

Generally, the easier a password is to remember; the easier it will be for an attacker to guess [61]. Even difficult passwords may reduce the security of a system [62]. This is because the user usually uses some password management strategies that weaken overall security such as writing down or electronically storing the password or re-using the same password for different services [63]. The more restricted requirements for password creation are the greater the degree to which users will adopt practices that subvert the security of the system.

The problem in this study is the need for a new authentication approach to strengthening the security and privacy of mobile cloud computing. In this work, our focus is to find answers to the following research questions:

How to strength the user's security and privacy by combining username/password with biometric (fingerprint) authentication in MCC?

What is the advantage of using a combination of username/password with biometric (fingerprint) from the existing authentication?

How to extend a combination of username/password with biometric (fingerprint) to improve security and privacy of user authentication in MCC?

1.4 Objectives

The general objective of the thesis is described as follow:

1.4.1 General Objective

The general objective of this thesis is to enhance user authentication in mobile cloud computing and propose a new authentication security architecture to enhance security and privacy in MCC.

1.4.2 Specific Objectives

To achieve the general objective of the study, the following specific objectives are identified.

- ✓ To study the authentication techniques currently available, especially those based on biometrics (fingerprint)

- ✓ To study the username/password trends currently available
- ✓ To propose an effective techniques, by combining username/password with biometrics (fingerprint) methods
- ✓ To propose adequate usable with providing a stronger security than the traditional solutions.
- ✓ To protect users data from unauthorized access, disclosure, modification and monitoring in MCC environment
- ✓ To discuss the strength and weakness of the propose technique, in the context of limited resources available on mobile device.

1.5 Methodology

The research was structured to be design research. The paper followed the research method suggested by Vaishnavi and Kuechler [62]. The research method considers the process of such research as a *cyclic research process* consisting of several steps or stages namely, *awareness of the problem, solution suggestion, artifact development, evaluation,* and finally *conclusion or result*.

Awareness of the problem: the existing (password or PIN) authentication and identity management were found to be vulnerable to various attacks. Current state-of-the-art solutions include methods for authenticating users by combining username/password with a fingerprint authentication method. The combination of username/password with fingerprint authentication-based authentication scheme is one of these solutions aimed at strengthening the security of online authentication for cloud services.

Suggestion: security of such a solution needs to be investigated. The usability of a username/password with fingerprint authentication in online banking and other online transactions needs to be examined.

Development**:** the practical solution aimed to improve the security of using MCC from the cloud service provider. In this solution, the security details validation process is done and tested by using Firebase.

Evaluation: the evaluation is done by selecting the most common attack types from both of the authentication methods. For the username/password we select out two most common attack types i.e brute force attack and keylogger attack types and from the fingerprint authentication we select Attack effort for lifting latent fingerprints from the surface and Fingerprint attack effort for fabricating dummies then we evaluate by using common evaluation methodology (CEM).

Conclusion: the result and the outcome of the evaluation were described in detail. A display assurance architecture that strengthens the security of username/password with fingerprint authentication scheme was the result of this work.

The proposed solution is entirely a new framework architecture which addresses authentication issue in mobile cloud computing as per the review conducted. The security model framework is presented using a unified modeling language (UML) drawing and IDE android tools.

1.6 Scope

The main scope of this work is restricted to the field of mobile cloud computing and security; and particularly on the MCC *authentication* field.

The other scope of this work includes only technical and architectural designs (framework) of the MCC authentications on the enhancement or strengthens of security and privacy problems, but servers-side attack is not our research focus and it's beyond the scope of this work.

1.7 Organization of the Study

This thesis paper is structured as follows: Chapter two presents a discussion about literature reviews such as: Cloud computing, Mobile cloud computing and about the security and privacy in MCC. Andin the other section of this chapter we review related works which is focus on the MCC authentications in detail. Chapter three is focused on Security in Mobile Cloud Computing and Authentications. It continues by presenting about security in mobile cloud computing and also focuses on the authentication issues in cloud computing. Chapter Four covers the proposed architectural security designs. Finally, Chapter Five is all about conclusion, recommendations and future work.

CHAPTER TWO

LITERATURE REVIEW AND RELATED WORK

In this chapter we cover two sections, in the first section we focus on the literature review which deals with the discovering concepts and theories on MCC authentications; and in the second section is the related work. In the literature review and related work sections, we focused on exploring different studies, results, and limitations of existing studies done related to our proposed work.

2.1 Literature Review

This section covers the technologies used basically in Authentication on mobile devices, cloud computing, mobile cloud computing, architectures of MCC, cryptosystems, Security, and Privacy in MCC and Authentication methods in Mobile Cloud Computing.

2.1.1 Authentication on Mobile Devices

Authentication is the first-line protection in any computer system or device as it is a requirement for several other security services such as authorization. From the point of view of a mobile device, authentication is typically achieved using a knowledge-based authentication method, and with this method, the user confirms his identity by demonstrating knowledge of the secret. This secret could be a character-based password, a digit-based passcode (also referred to as Personal Identification Number (PIN)) or a graphical-based passcode.

Most security applications are designed based on knowledge or token. Knowledge-based applications authenticate identity by checking ’something you know’ such as a PIN, password, and so on. Token-based applications check ’something you carry’ such as a key or card. There are fundamental flaws with these two types of security mechanisms. Knowledge such as passwords and PINs can also be easily forgotten or guessed using social engineering [7] or dictionary attacks [8]. Similarly, tokens like keys or cards can be stolen or misplaced.

Biometrics technology provides a more feasible and reliable mechanism based on ’who you are’. It identifies people by their physical personal traits, which inherently requires the person to

present at the point of identification. Biometrics refers to the statistical study of biological phenomena, such as the physiological features and behavioral traits of human beings [64]. The physiological features can be a fingerprint, hand geometry, palm print, face, iris, ear, and etc. Using password/PIN or biometric authentications is not guaranteed and secured way, so we proposed a new authentication approach which is a combination of username/password with fingerprint authentication.

Using a single authentication method to form an authentication system brings in several limitations that may hinder usability. For example, a face-based authentication method imposes a higher privacy risk than other biometrics-based authentication methods as it can be more likely used in a covert manner [10].

One of the possible measures to make the authentication service more secure, while, at the same time, without hindering usability, is to integrate a biometric-based (e.g. fingerprint) Authentication method with a knowledge-based authentication method. These types of authentications are called two-factor authentication systems. Such a system can make unauthorized access to mobile devices harder, thus strengthening the security protection level of mobile devices, while, at the same time, still be able to maintain usability as high as possible. The existing knowledge-based authentication method has a wide social acceptance, and the fingerprint authentication method is also expected to be widely accepted by the general public [12].

2.1.2 Cloud Computing

Cloud computing is an evolution in the field of computer science and technology. In the twenty-first century, computer users access Internet services via lightweight portable devices because powerful desktop machines are going through a phase of drought. Cloud computing emerged as a solution to this problem. Cloud is a distributed computing paradigm. It is a collection of interconnected and virtualized computers, which are provisioned and presented dynamically as unified computing resources offered on a pay-per-use basis [13]. Cloud computing is defined as applications that are delivered as Internet services: the hardware and system software in the data centers are used to provide these services [14]. Cloud computing is an advanced technology that focuses on the way of designing computing systems, developing applications, and leveraging

existing services for building software [15]. It is based on dynamic provisioning. In cloud computing, resources are offered in an on-demand and pay-per-use basis from the cloud computing vendors [15].

“Cloud computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services” [16]. With regards to cloud services, the cloud computing types can be divided into two groups: *Service Models*: Service model refers to the services type that provided by cloud providers such as Amazon EC2 and Microsoft Azure etc. Software as a Service (SaaS) refers to delivery of software applications or services that run in cloud provider infrastructure. The consumer does not control or manage the resources like servers, operating system and storages. Platform as a Service (PaaS) refers to delivery of deployment services onto cloud provider infrastructure whereas the consumer can deploy certain applications with limitations on control on management of the resources. Infrastructure as a service (IaaS) refers to delivery of infrastructure provisioning services to the consumer where the consumer can provision servers, operating system, storage and others fundamental resources [15, 18].

Deployment Models: Deployment Model refers to how cloud infrastructure owner host and offer the cloud computing services. These are public, private and hybrid cloud models [18]. Public Cloud model refers to consumer usage of an organization’s cloud infrastructure on a pay per use basis. And also a public cloud is shared and used by customers via the Internet; for example, Amazon Web Services is the leading public cloud provider. Private Cloud model refers to internal deployment, management and operation of an organization’s own cloud infrastructure. It is a network of all services or a data center that stores hosted services for a restricted number of users.

Hybrid Cloud: model refers to integration on both public and private cloud models within an organization.

2.1.3 Mobile Cloud Computing

Mobile cloud computing is defined as rich mobile computing technology. This technology controls integrated elastic resources of different clouds and network technologies toward

unlimited functionality, mobility, and storage to serve a large number of mobile equipment anywhere and at any time through the Ethernet channel or Internet. This helps to despite heterogeneous environments and platforms based on the pay-as-you-use principle [13]. MCC is an infrastructure where the data storage and data processing are performed outside the mobile device but inside the cloud. In MCC, the computing power and data storage are moved away from mobile devices and performed in the cloud, bringing mobile cloud applications and mobile computing not only to smartphone users but also to a wider range of mobile subscribers [14]. So, MCC is an infrastructure that combines the mobile computing and cloud computing domains where data storage and processing happen outside the mobile device. It is not always that offloading will be to a remote cloud, but it can be local cloud storage or shared resources of nearby mobile devices.

Integrating mobile applications with cloud environments have recently become a popular approach in the industry. The term mobile cloud computing means to run an application such as Google's Gmail for Mobile on a remote resource rich server (in this case, Google servers), while the mobile device acts like a thin client connecting over to the remote server through 3G [13].

Mobile devices are typically challenged in terms of processing power, storage capabilities and communication issues. Although mobile computing provides users with mobility that allows them to accomplish various tasks on the mobile cloud computing because it provides the benefits of both mobile device and cloud computing. In a mobile cloud computing environment, mobile device's data storage and data processing are outsourced to the cloud. This increases battery life, computation processing, sharing of information as well as running computation intensive applications on a resource constraint mobile device [14].

Several approaches and techniques have been proposed to augment mobile devices by leveraging cloud computing. However, trust and privacy are still a major issue in MCC that hinder its vision. SAMI [19] is an arbitrated multi-tier infrastructure model for mobile cloud computing leveraging Service Oriented Architecture. Authors [20] present Focus Drive, a new mobile cloud data processing framework through trust management and private data isolation. Authors [21] provide a trust model for offloading computation from mobile devices to a public cloud network. The main precursor for this research is improving mobile device energy efficiency. Zhang et. al.[22] provide the concept of personal cloud network. They propose a novel approach of using

hybrid cloud to distribute the computing among mobile devices, private cloud and public cloud. User data is stored within the personal cloud framework where users have full, physical control. User can authorize public cloud services to access user-approved data.

2.1.4 Architectures of MCC

Mobile phones are now ubiquitous in most people's daily activities and this has motivated companies to develop applications that can be easily accessed through mobile phones. The Internet, GPS and games applications are behind the worldwide popularity of mobile devices. However, limited resources (e.g., CPU, memory, and data storage) of mobile devices pose some design challenges to mobile application developers. To overcome these challenges, cloud computing is being used. Mobile cloud computing combines the concepts of cloud computing and mobile computing [24] [25]. This new technology makes use of the capability of data storage and data processing by using cloud computing infrastructure through the Internet. There are many MCC architecture is developed from various stages, but this is the general architecture of MCC.

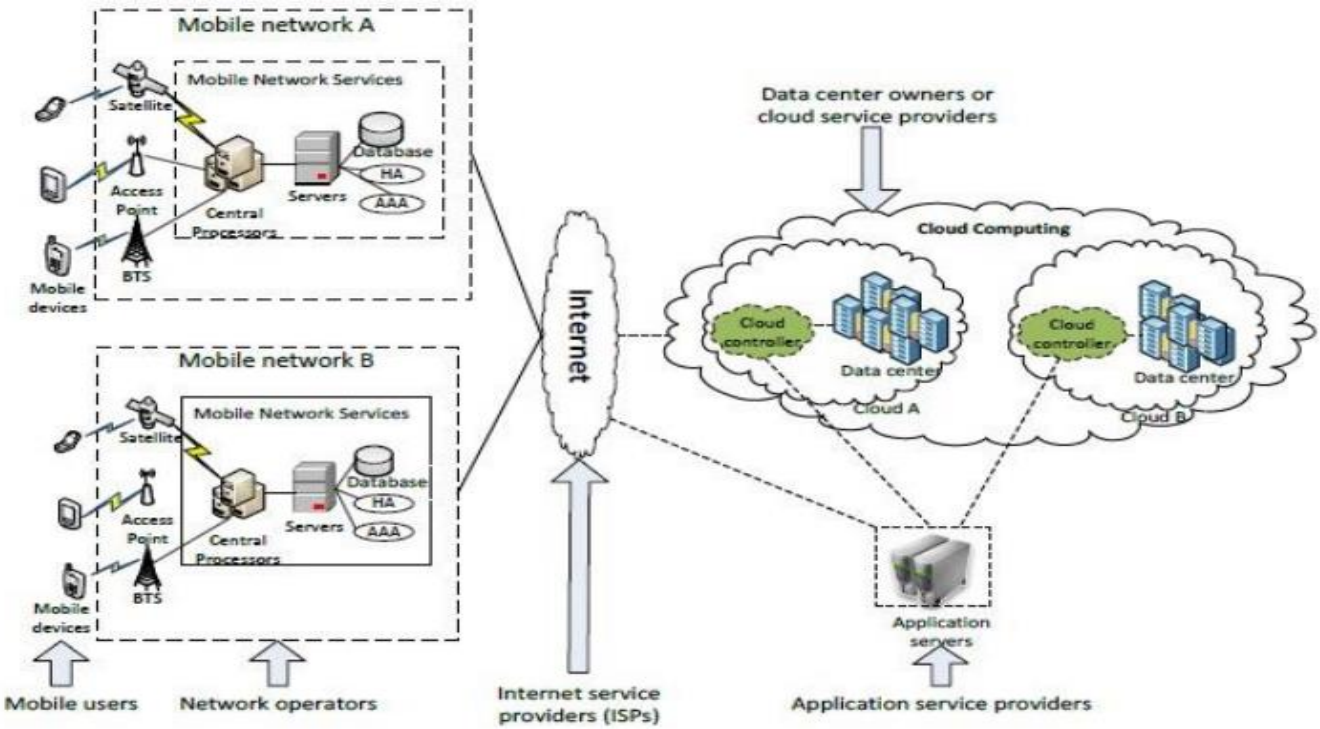

Figure: 2.1Architecture of Mobile Cloud Computing, Source: Dihn et al., 2011 [2011]

In this architecture the mobile devices that are linked to mobile networks through base stations (Base Transceiver Station, access point or satellite) which connects and control the functional interfaces between networks and mobile devices. Mobile user's data requests and information are transmitted through central processors from servers connected to mobile network services. Mobile users gets services as AAA (Authentication, Authorization and Accounting) from mobile network providers based on the home agent and user's data that is present in the databases. The users requests are send through internet to cloud. In the cloud these requests are controlled through cloud controllers and provide services according to their request [25].These services are achieved with the utility computing, virtualization and service oriented architecture.

2.1.5 Crypto systems

The cryptography system is a combination of three cryptographic algorithms these are: asymmetric, symmetric and hash function.

2.1.5.1 Symmetric Algorithms

Symmetric algorithms encrypt and decrypt a message using the same key or a shared secret key. The most popular algorithms are DES, 3DES (Triple DES), and AES. The first two of these algorithms are generally considered obsolete, but AES is the standard symmetric algorithm. Therefore, our proposed framework used the AES algorithm for the sensitive and secure data storage on cloud environment.

Table 2 1 comparisons of the performance of AES, DES and DES3

Factors	**AES**	**3DES**	**DES**
Key length	128,192 or 256 bits	(k1,k2 and k3)168 bits (k1 and k2 is same 112 bits	56 bits
Cipher type	Symmetric block cipher	Symmetric block cipher	Symmetric block cipher
Block size	128,192,or 256 bits	64 bits	64 bits
Developed	2000	1978	1977
Cryptanalysis resistance	Strong differential truncated differential ,linier interpolation and square attack	Vulnerable to differential, brute force attacker could be analyzed a plain text using differential cryptanalysis	Vulnerable to differential and cryptanalysis; weak substitution tables
Security	Considered secured	One only weak which is exit in DES	Proven inadequate
Time required to check all possible at 50 billion keys per	For a 128-bit key:$5*10^{21}$	For a 112-bit key :800 days	For a 56-bit key :400 Days

AES (Advanced Encryption Standard)

The Advanced Encryption Standard (AES) is a symmetric-key block cipher published by the NIST in December 2001 [26]. AES was designed after DES. It is a non-Feistel cipher that encrypts and decrypts a data block of 128 bits. The key size, which can be 128, 192, or 256 bits, depends on the number of rounds. AES has defined three versions, with 10, 12, and 14 rounds, each round keys are always 128 bits [26]. Most of the known attacks on DES were already tested on AES [26]:

- ***Brute-Force Attack:*** AES is definitely more secure than DES and other block cipher algorithms due to the larger-size key.

- ***Statistical Attacks****:* Numerous tests have failed to do statistical analysis of the cipher text.
- ***Differential and Linear Attacks:*** There are no differential and linear attacks on AES as yet.

Over-all the algorithms used in AES are so simple that they can be easily implemented using cheap processors and a minimum amount of memory.

2.1.5.2 Asymmetric Algorithms

Asymmetric algorithms use a different key to encrypt than they do to decrypt. The encrypting key is called the *public key* and the decrypting key is the *private key* **[26].** These algorithms can also work in opposite direction. There are three asymmetric algorithms in use today: Diffie Hellman, RSA, and ECC. Diffie-Hellman is not quite suitable for establishing identity as describe above, but the other two are. RSA is the most commonly used today, but ECC appears to be on its way to becoming the next standard. Unlike symmetric, asymmetric algorithms are limited in size of message that it can encrypt and decrypt.

Elliptic Curve Cryptography (ECC)

ECC is a public cryptographic scheme that uses the properties of elliptic curves to generate cryptographic algorithms. In the 1980s Koblitz and Miller proposed using the group points on an elliptic curve defined over a finite field in discrete logarithmic cryptosystems **[26]**. The encryption/ decryption techniques use the properties of elliptic curve to generate the key pair instead of using the product of two very large prime numbers in RSA. Elliptic curves are the binary curves and are symmetrical over x- axis. These are defined by the function **[27]:**

$y2 = x3 + ax + b$ eq. (1)

The above eq.(1) shows that an elliptic curve graph like the other mathematical graphs, but the only difference is that drawing a tangent line on the graph cross the graph in to two points. By counting the number of crossing from y axis to the x-axis we calculate the secret key (private key) of ECDSA for signature. ECC includes the elliptic curve Diffie-Hellman key exchange, elliptic curve ElGamal public key encryption, and the elliptic curve digital signature algorithm.

2.1.5.3 Hash Function

If we are using asymmetric algorithms to sign, we must first calculate a digest, a smaller number based on the larger message by executing a hash function. Some hash functions were invented for error detection during transmission. These hash functions are not suitable for digital signatures because they are easily reversible. Instead, we devised cryptographically secure hash functions, which produce hashes that are difficult to reverse. In other words, given a hash, it is difficult to create a document that calculates that hash. These hash functions include: MD5 family, SHA 1, SHA 2 (SHA-128, SHA-192 and SHA-256) and SHA 3 family. For security and performance issues, our proposed system has dealt with an algorithm of lightweight security, AES, SHA-1, SHA-2 and ECDSA.

2.1.6 Security and Privacy in MCC

The data security and privacy protection are the important issue in the mobile cloud. First of all, in the cloud the ownership and management of the users' data are separated, which cause that the worries of users to their own information resource become the important obstacle for the popularization of the mobile cloud computing. In addition the users' data are stored randomly in the shared infrastructure all over the world, and users do not know the specific position in which their data are stored. So users' private information faces increased risk of exposure [28].

The authors in [29] have provided comprehensive information regarding the cloud security problems. The authors inspected the security problem from cloud architecture point of view, the cloud stakeholders' point of view and at the end from cloud service delivery models point of view. From architecture prospective, the cloud service providers need to provide multi-tenancy and elasticity as both these characteristics play a major role in cloud security. From stake holder prospective, the security configurations needs to be organized so, each service should be maintained a level and at runtime. From service delivery model prospective, the IaaS, PaaS and SaaS models have security issues. The cloud management security issues and cloud access method security issues are also highlighted.

The authors of [30] have presented an overview of MCC security architecture. Privacy and integrity of the data is important aspect of MCC security. The author categorized the users' in term of security into two categories: mobile network security and cloud security. In first

category the security for mobile applications and privacy are explained. The second category is about securing the information on the cloud or simply securing the cloud. In cloud security the authors highlight very important concerns associated with data integrity, authentication and digital rights.

The authors in [31] have provided details about the security issues which cloud service providers are facing when they dig deep for cloud engineering. Therefore, in order to ensure data and application security in cloud environment, the cloud service providers must follow the Manages Service Model (MSP). A detailed survey results which is conducted by International Data Corporation (IDC) highlights that security is the biggest concern of IT executives and other peoples involved in an enterprise's decision to move for cloud services. There are some serious issues and challenges which cloud computing are facing in the domain of cyber security.

The paper [32] presented a detailed analysis of data security and privacy protections issues along with the existing solution to provide protection against these issues. Authors supported their arguments by the surveys from IDCI and Garter. Detailed cloud security architecture has also been explained. The security architecture highlights the infrastructure, platform, software security along with the services related to auditing and compliance. Cloud computing is facing serious data security and privacy issues which need to be addressed.

In [33] the authors have identified the serious threats and risks related to privacy and security for the mass and corporate users when they will integrate their mobile hand held devices with the cloud infrastructure. The paper points towards the different motivational factors which are forcing mobile cloud operators to move their services and operations to cloud. Some of the key motivational factors are business interest, user demand, preparation of network service provider, QoS and mature technologies. The authors conducted a survey that how wireless mobile devices integrates with the cloud. The people targeted for the survey are mobile device users, cloud developers, IT manager or executives and wireless network administrators. These people are targeted in order to get proper results whether the security and privacy concerns of the users have increased or not if they are planning to move for the cloud.

The general security requirements for MCC can be derived from the security requirements defined by ITU [34] and US National Security Agency [34], which are summarized in the following.

1) *Confidentiality*: The confidentiality is fundamental requirement that refers to keep mobile users' data secret in the cloud. Here, the confidentiality is a big hindrance for mobile users to avail the cloud services. As the data is transmitted and received within public networks, and stored or processed in public cloud servers to avail the cloud services, there is possibility to reveal the data to unauthorized parties.

2) *Integrity*: In MCC, the data storage and processing are resided on the service provider's end. Here, the integrity needs to ensure the accuracy and consistency of users' data. In other words, the integrity prevents undetected modification of the data by any unauthorized users or systems. The violation of integrity affects the mobile users in their business, economic and other losses.

3) *Availability*: For MCC, the availability ensures that all cloud services must be available always at any places as per mobile users' requirements. Ensuring availability includes preventing different kinds of availability attacks which make delay, alter or interrupt the availability of services.

4) *Authentication and Access Control*: The authentication is the process or act of determining the identity of a user, user's data or application. After successful authentication process, it is needed to determine what resources are permitted to access and what kind of actions can perform such as view, run, modify or delete. This is called access control.

5) *Privacy Requirements for MCC*: The security objectives such as confidentiality, integrity and authentication persuade the privacy and these objectives preserve the privacy directly or indirectly of the cloud service users in mobile devices.

2.1.7 Authentication method in Mobile Cloud Computing

In the cloud computing environment, users use an authentication system to utilize the cloud services through a Web-based user interface, either a web browser or a mobile application, or a web service application programming interface (API). Authentication on the cloud is necessary

to provide secure access to the cloud services by authorized users only. At present, authentication is done in different methods, such as a simple text password. The transmission between mobile device and cloud server has taken once they authenticate each other; this ensures the second communication between two legitimate parties.

Authentication is the process of comparing the credentials with the registered credentials which are stored in the database and if the matching is successful then user is allowed to proceed. If the matching fails the user is not allowed to proceed and is informed to provide valid credentials [35].

There are many types of authentication, such as device based authentication, password based authentication, and biometric authentication.

Device Authentication: Hardware devices which are portable and similar to smart cards or credit cards containing magnetic strip is used for authentication. For example, ATM cards with the magnetic strip on the backside of the card are used as an authentication to access bank account. If the user gives correct pin then it allows him to use the user account if not it blocks the card [35].

Password Authentication: The use of password is frequent form of usage for authentication. This method requires that each user must have a user name and a password. It requires the user to provide the password with the specific characters. This technique is cheaper and its use is most widespread [36].

Biometric Authentication: The features used in this approach to authentication cannot be stolen, forgotten, or forged. For example, a user's fingerprint, eyes, face, hand structure, or voice cannot be tampered. These are the features which are most helpful for authentication because every human being has his own characteristic features which cannot be forged or stolen [36].

Importance of Authentication: Mobiles have become part of every individual's life. The integration of mobile applications with the existing cloud computing is called Mobile Cloud Computing (MCC). MCC offers many advantages for the user which are related to the use of available platforms, services, and networks. Mobile devices have access to large amount of storage in the cloud for storing data. Since resources in the cloud are shared, it is essential that

individual user's data is secure and is inaccessible to unauthorized users. One of the most effective ways for providing security is authentication of a user before he is given access [36].

In the article Mobile One Time Passwords and RC4 Encryption for Cloud Computing, Johnson et .al. in [37], authors proposed policies for securing data transmission over the internet, wide area network and authentication. RC4 encryption algorithm was used to ensure the security of data as this algorithm is secure enough and fast to be run. Compared to AES, which together with RC4 are the most common encryption methods are being used over the internet. Authentication phase achieved by using one time password for the login to cloud computing services by being generated on regular mobile phone as a third party to confirm the client credibility before allowing the client to access cloud computing services. The difference here, with static passwords is that the passwords in the proposed method are only valid for one time only to ensure more secure. All data exchange among the server, clients and mobile agents will be encrypted by RC4 algorithm.

Now there are two kinds of authentication approaches which attract significant attention. The one is user-centric identity authentication. In this approach, a user is identified and defined through identifiers or attributes, and a user can be allowed to have multiple identifiers. By this way we can research a desired user-centric identity management mechanism for mobile clouds [38]. Here in our work focus on by combining username/password with biometric (fingerprint) approaches to solve the security issues in the MCC authentications.

2.2 Related Work

This section covers the existing MCC authentication related works and see in detail the strength and the weakness of the papers related to the MCC authentications.

In the cloud computing environment, users use an authentication system to utilize the cloud services through a Web-based user interface, either a web browser or a mobile application, or a web service application programming interface (API). Authentication on the cloud is necessary to provide secure access to the cloud services by authorized users only. At present, authentication is done in different methods, such as a simple text password.

In a traditional authentication technique, users need to enter multiple keys for accessing a service from a system, e.g., username/password or PIN, which is verified by the system using its database. The process is repeated by the user for accessing a different service or when the session expires due to user mobility or network failure. Currently, most CSPs use traditional authentication for providing their services [39]. However, the authentication process incurs significant delay as the keys are entered manually (sometimes using mobile devices). Thus, the traditional authentication technique is not efficient for accessing on demand services from multiple CSPs. This is more significant for MCC, as switching between various clouds based services or changing network condition requires re-initiation the authentication process, incurring significant overhead on the mobile devices.

There are several proposed strong user authentication provided by researchers to improve mobile cloud security. Omri et al. in [40] introduced an application that uses handwriting recognition as an authentication system to secure access in mobile cloud. In this way, the user is identified by password and unique handwriting style. This application, which used the mobile phone as a biometric-capture device, also used Hadoop[1] to establish the connection between mobile user and the cloud via Internet. It has been implemented into two ways. The main difference is in the implementation mode, one as web page and the other as mobile application. The limitation in this proposed application is that the person's handwriting may change due to different reasons.

[1] *Apache Hadoop is an open-source software that provides applications both reliability and data motion*

A biometric authentication mechanism that uses fingerprint recognition systems to secure mobile cloud computing was proposed in [41]. The proposed authentication mechanism uses existing cameras in mobile phones to capture the fingerprint image of a cloud user. Hence, this mechanism does not require any extra sensors to be implemented into the mobile device. However, a high-quality camera is required for capturing an accurate fingerprint image for operations to be carried out. The user captures a fingerprint image using a mobile phone camera. The fingerprint image undergoes image processing such as converting an RGB image into a gray-scale image, reduced blurring, segmentation, and ridge enhancement. The processed image is then sent in a core-point detection phase where feature extraction of the fingerprint image is carried out. Finally, the cloud server checks whether the extracted fingerprint image matches the one that is stored in its database. If it matches, the user is verified and authenticated for the cloud server. Here the user's phone device must have a high-quality camera to capture a fingerprint if the device does not have a high-quality camera then cannot be authenticated because of the capture fingerprint image.

In [42], suggested using quick response code (QR code) for a user authentication system in the mobile cloud. In this system, the user ID, password, and the user's image are converted into QR code. In the multilevel authentication system proposed in [43], this authentication system generates and authenticates the password at multiple levels to access the cloud services. Access to the cloud is allowed if authentication is successful in all levels.

- First level of authentication is the organization level. This level reads the organization password; if unauthenticated they are going to terminate. If it is authenticated, then it enters a second-level authentication.
- Second level of authentication is the team level. This level reads the team password; once authentication is done, it then enters a user-level authentication.
- Last level is the user level. This level reads the user password to provide the user privileges and permission.

The balance between security and usability must be found [44], just as when trying to apply the authentication system from a traditional client server to cloud computing. It can be applied, but it has a high risk because the infrastructure in the cloud is shared among users and managed by the cloud providers.

Ziyad and Kannammal [45] proposed a multifactor biometric authentication system for cloud computing environment. This biometric method was fingerprint and palm vein. The goal was to handle the biometric data in a secure fashion by storing the palm vein biometric data in multi-component smart cards and fingerprint data in the central database of the cloud computing security server.

In this proposed technique, the processes of matching biometric data were performed on the card with Match on Card technology; therefore it helped improve security system.

H. Sun, et, al [46] proposed identity based (ID) user authentication and key agreement schemes for the mobile client-server environment. The proposed system was subject to several drawbacks such as the scheme was suffered from design weakness. The server knows all the users private keys. The proposed scheme was more suitable for the mobile client server environment. *A. A. Yassin* [47] proposed a new setting, where users do not register their passwords to the service providers. The passwords were supplied with necessary information from the data owner to the service providers. The proposed method was more suitable for the cloud environment and withstands different known attacks.

Chandra ShekharVorugunti [48] has introduced a new concept of BioAaaS to maintain secure authentication. Based on SAAS model of Cloud it provides a light weight and secure authentication mechanism. It contains two steps for authentication. First is Enrolment and next is Verification. In Enrolment process the biometric data is converted into a binary form. The feature extractor then converts the binary string into a set of features. In verification process same process has processed when the user logins to the cloud. The matching module matches the features of the stored data and login data. Thus they have provided a service to do heavy weight cryptographic encryption and decryption operation on user's biometric data.

A private authentication system proposed in [49] uses a Smart Card Generator (SCG). The proposed system uses a dynamic nonce generation and bilinear pairing cryptosystem techniques. According to the researchers, the technique reduces the complexity of discrete logarithm problems. To set up the authentication system, the SCG selects a random number as a master private key and computes a public key to generate all other public parameters. It then publishes the generated public key and public parameters. The registration phase is executed after system

set up is complete. Mobile users or service providers register to the SCG by providing their information while the SCG computes and securely sends the respective private keys to the mobile user and the service provider. When the service provider and mobile user want to communicate, a card provided by the trusted SCG is used to authenticate both parties. The disadvantage of this system is that there is a risk of losing the card, and the card is necessary for both the mobile client and the cloud service provider to authenticate each other.

An authentication system called Mobile Cloud Key Exchange was proposed in [50]. This system is based on the randomness-reuse strategy and the Internet Key Exchange (IKE) scheme. The researchers discuss the various authentication technologies that are used by mobile clouds such as basic authorization, signature tokens, open ID, and open Auth (OAUTH). The authentication system is setup from a Diffie-Hellman group using a large prime integer and generator number for the group that acts as a primitive root modulo.

The system uses certificate authority in a public key exchange to ensure that communication between two parties can be verified. The MCKE system is used when it receives a new task so the cloud controller (CLC) pick a secret value, compute a public key, and broadcast the message to the domain of the server for a Diffie-Hellman key exchange to take place. The session key is now shared between the CLC and the server to encrypt their communications. For MCKE to be completed, CLC needs to generate signatures using a secret key generated from a key pair issued by the certificate authority. The signature will then be sent to the server for verification to take place. The public key contained in the certificate is used to verify the signature. Likewise, for verification in CLC, the server also needs to send its own signature, encrypted ID, and certificate to the CLC. According to the researchers, the proposed authenticated key exchange scheme reduces time consumption and computation load but is costly to implement. The researches mentioned that new strategies for providing better security-aware scheduling are required furthering enhancing symmetric-key encryption.

Lehab AL Rassan et al. in [41], proposed a model for Securing Mobile Cloud using biometric authentication (SMCBA). They used fingerprint recognition system to enhance mobile cloud resources, which consists of using the mobile device camera as a sensor to take the image of the fingerprint, then make a pre-processing of the image, feature extraction is realized after that, and at the end, compared it to the saved data for the authorization.

The major disadvantage of the proposed classical models is the manual extraction of the characteristics. This motivates us to use the combination of username/password with biometric (fingerprint) authentications to make reliable authentications in MCC.

The authors of [51] proposed a lightweight protocol for mobile cloud environment based on local mobile network authentication. The protocol was divided into two phases; registration phase, and mutual authentication phase, where the mobile user registered with their mobile service provider by entering their International Mobile Station Identification number (IMSI) and a personal secret. The service provider issued the user with an Authentication Certificate (AC) and the user used the secret key provided with the certificate to encrypt the login messages. Once the login was successful, a session key was established and used for further communications. The protocol was lightweight as it used only two-factor authentication, symmetric encryption was also low in computational costs, and it also has a low latency. However, the protocol did not establish a secure channel, which made it susceptible to attacks. The AC was stored on the mobile device so if the mobile device was lost, stolen, or compromised; the device could be used for impersonation attacks.

The authors in [52] proposed a real-time biometric recognition system that is based on Orthogonal Line Ordinal Features (OLOF) extraction technique. The processing of the authentication is distributed on the mobile device and the server. The use of palm print that can be easily captured provides more security than any other authentication methods. However, this system could lead to a delay in the authentication process due to the intensive processing that is carried out by the mobile device (i.e., the capturing of the image, the correction of alignment functions, and the collection of data in display image and transmission). Additionally, it does not protect the biometric data from exposure during the authentication process. This method showed that using biometrics to authenticate the user may affect a protocol's ability to be lightweight.

The authors of [54] have proposed a fuzzy vault authentication protocol. The protocol is based on digital signatures and zero-knowledge authentication. Asymmetric RSA Keys were used to provide authentication for the mobile device and the server. A fuzzy picture password method was used to provide authentication and usability, especially when users have a difficult time remembering a password required for sufficient length and randomness to be secure. Upon completing the authentication process, a secure communication channel is set up to connect the

mobile device with the cloud server. The server starts exchanging Diffie–Hellman (DH) public values with the client. This gives the protocol resistance to man-in-the-middle, impersonation attacks, reply attacks, and sniffing attacks. However, the high amount processing on the client side (mobile device) leads to energy wastage and less battery lifetime. Moreover, the existence of the fuzzy image data on mobile device could be exposed, exported, or copied in the case of device loss.

The authors of [53] also investigated the accuracy performance of touch dynamics using a digit-based input string. In their investigation, they recruited ten subjects. Each subject was asked to provide 100 input samples of a predefined 4-digit PIN. They used the Multi-Layer Perceptron classifier to classify the legitimate subjects, and these subjects are correctly classified up to 86% of the time. The results are encouraging, but to achieve the reported level of accuracy performance, they have made use of 100 input samples per subject to train the classifier, and this sample amount is considered to be large. Acquiring such a large number of samples from the subjects is time-consuming and not always practical during an enrolment phase. It is not clear if the same level of accuracy performance could still be achieved when a smaller number of input samples are used. With this approach, an equal number of the legitimate and the illegitimate samples can be obtained with minimal or no additional resources. However, this approach has a limitation when the input string is not the same across all subjects, as it is not possible to compare the touch dynamics patterns of two input strings when they are different.

CHAPTER THREE

THE PROPOSED HYBRID AUTHENTICATION TECHNIQUE

The main intention of this chapter is to discover a clear image of existing framework and design of proposed system frameworks focusing on strengthening the security of mobile cloud computing using hybrid authentication techniques.

3.1 Overview of the Proposed Solution

The proposed scheme incorporates the username/password with biometric (fingerprint) as credentials repository. Each user uses a set of credentials per required authentication. The credentials usually consist of username and password followed by the fingerprint. The set may be altered in order to acquire a new password, yet the username and the fingerprint cannot follow that procedure as services provided to user from authentication authority are linked to that (user name and fingerprint). A set of passwords may be linked to each username and fingerprint in order to change the credentials set dynamically. Through this method a given set of passwords is pre-computed and linked to the given username and fingerprint by the issuing authority. In such a way, issuer and user and fingerprints are synchronized regarding the authentication credentials required. As the authentication procedure may choose a set of credentials through the potential combinations of the pre-computed passwords, there is no need for password renewal with the intervention of user or any other system outside the secure repository environment of user name and fingerprint. In this way no one except the issuer-authentication service may have knowledge of the actual value of the identified password.

The proposed solution is taking advantage of the architecture of strengthening the security of user's authentication. Furthermore a username/password with fingerprint consists of a computational system that maintains secrecy as it becomes operational only when plugged into an appropriate hosting device while at the same time it Maintains information in multiple security layers.

3.2 Fingerprint identification process

In the identification process the user doesn't need to state who he or she is. A username/password and a fingerprint is taken and compared to each username/password and fingerprint in the database of registered users. When a match occurs, the user is "identified" as the existing user of the system found. Since the newly acquired username/password and fingerprint is compared to many stored username/password and fingerprints, this is called a one-to-many matching process. As in the verification process, when username/password and fingerprint identification is done, only the fingerprint template is used in the comparison, not the actual image of the fingerprint.

3.3 Username and Password

Most of the mobile applications require users to register with the username and password to use the MCC in the clouds. A user sends his/her username and password to the network authentication server for authentication. The authentication server verifies the user by his/her username and password in the server’s database. If it's matched with previously stored, the server can authenticate the user to access the cloud services otherwise the authentication server displays error messages to enter the correct username and password. Using username and passwords are not enough to access the cloud services; because it’s more exposed to attacks and frauds, so it’s better to use a hybrid authentication method i.e combining username/password with fingerprint authentications.

3.4 Proposed Solutions

By using the combination of the username/password with a fingerprint authentication method, the proposed technique consists of the two major phases. These phases are described below:

3.4.1 Registration phase

Personal Detail Acquisition: in this step, it is necessary to register a new user by requesting the user to fill the registration form contains user personal information which helps to identify the users account in a particular database (authentication server). When the user wants to register after filling the registration form, the user asks to give his/her username and then the system check for the redundancy of the username, if the newly added username is already registered with the existing username, then the system can automatically asks the user to change or enter another username, If the user username is accepted by the system, then the user require to give a password for his/her username. Then the system creates the username with password and then save the information on the system. Once the username is created and saved to the cloud (on the authentication server), the next process is the registration phase is to capture the fingerprint of the user.

Capturing: - in our proposed technique, by using a fingerprint button of the smart mobile phone the user fingerprint capture. After capturing the user fingerprint features are extracted by using appropriate algorithms approach and latter it is analyzed and converted to a template and are stored in the Cloud service Provider's authentication server.

Processing: - in this process the fingerprint image is processed i.e. this process considers some features which are required and eliminates the other.

The registration phase

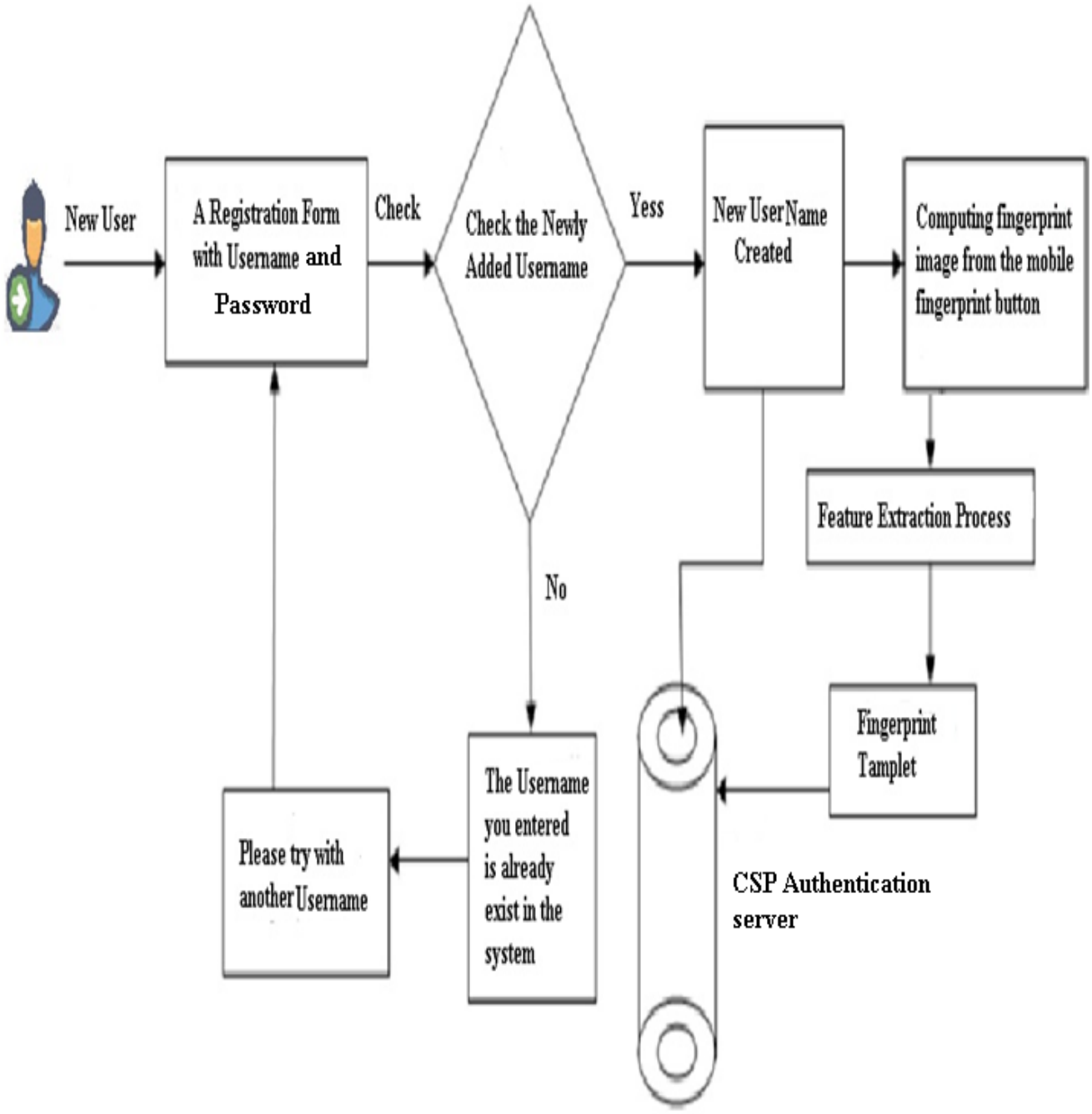

Figure: 3. 1 Registration phase

Feature extraction: -is done after the preprocessing to increases the accuracy by extracting features from the processed fingerprint. The preprocessed image is sent for feature extraction to the service which is provided by cloud

This is the process how registering phase works, how features are extracted from the fingerprint image and stored in the database for later authentication.

3.4.2 Authentication Phase

The second phase is authentication phase, whenever the mobile user wants to use the services of the cloud, the authentication must be done before using the system.

Personal information Entry: - In this step, user should enter his/her username with his/her password first. Then the username he/she entered checked from the authentication server or from the database and if there is a match, then the system prompts the user to authenticate. When the username match was not found, then the system prompts the user to enter his/her valid username and password. This process is done for a maximum of three times, if the user tries to enter the wrong username and password then the system automatically locks the user account and reports to the owner of the user account.

Capturing: - The user has to press the fingerprint button from his/her phone button. Then the image is sent to the service provided by the cloud for the next level process (authentication).

Preprocessing: - The fingerprint which is received from the users mobile is preprocessed which mean used features are highlighted and the remaining image is cleared.

Features Extraction: - The preprocessed fingerprint is sent to the service provided in the cloud for feature extraction, then by using the algorithms features are extracted and converted to a fingerprint template which is store in the database.

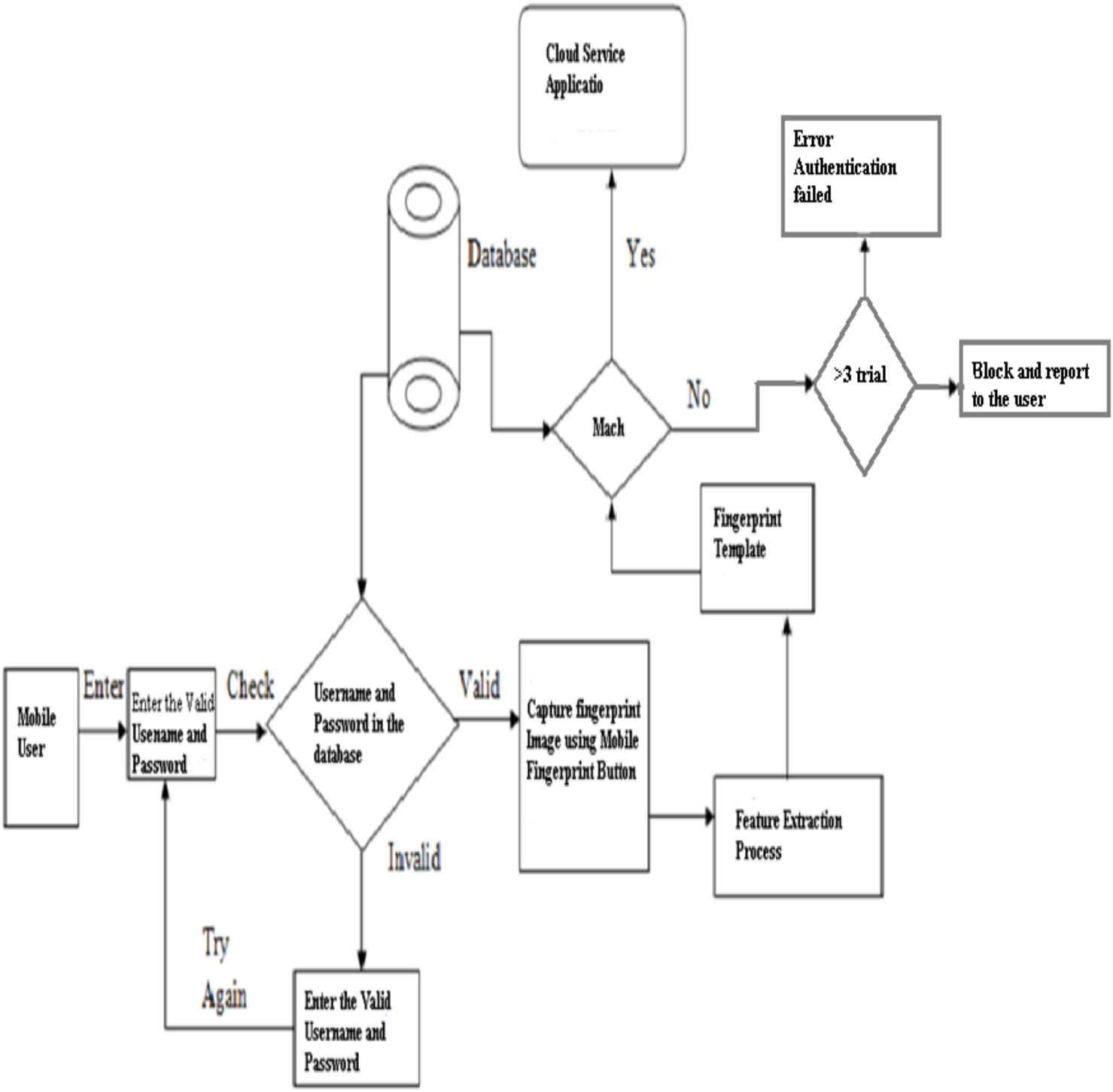

Figure: 3. 2 authentication phase

Comparison: - If the converted fingerprint template matches with the fingerprint template stored in the database or in the authentication server, then the user is authenticated to use the services of the cloud and if the match was not successful then the user is not allowed to login to the system and he/she cannot use the cloud service and gives as an error authentication messages.

3.4.3 Illustration of Proposed Technique with an Example.

Here taking an example of a user uploading his/her fingerprint into his/her cloud service provider's database, when the user wants to process any transaction or to do any task, then the first step is to authenticate his/her identity. The system prompts the user to enter his/her username/password first, and then it matches the username/password with the username stored in the system and if there is a match, then the next step is the user's fingerprint is captured as the next-level security. The captured fingerprint will be processed and the features are extracted and then converted to a template. The template will be matched with the template stored in the database and if there is a match, then the user is authenticated to use the services of the cloud as his/her interests. Consider one more example i.e. if the user wants to do some bank transaction (e.g. transferring money from one account to another) by using a mobile cloud service, then the user asks to authenticate by the cloud service provider and this is the place where authentication helps. Now by using the username/password with a fingerprint process the user can have a secure transaction.

The same process continues in this example also. The user needs first to register in the cloud account and when the user wants to do some tasks on the cloud, he/she requests the cloud service providers authentication server and the cloud service providers authentication server application asks the user to give his/her username and password. When he/she gives his/her username/password, the cloud checks for the matching in the authentication server, and when matching is successful it asks to give him/her the password, and now as a password user needs to give the fingerprint using the fingerprint button from the smartphone. Then the fingerprint button captures a user fingerprint and then features are extracted from the fingerprint image. Then it is converted to a fingerprint template and now this template is compared with the registered template and if the matching is a success. It allows the user to transfer the amount from his bank account.

3.5 Design of Security Solutions

The design of the username/password with a fingerprint-based security system is to improve the existing security system. The widely accepted methods, such as passwords, tokens, keys, etc., do not guarantee a high-security system, as they can be forgotten, guessed, stolen, donated, duplicated, or shared [61]. Preparing a secured two-factor authentication is a crucial thing in the cloud computing environment. In a security system design, the choice of verification or customer identification for the image comparison procedure is very important.

3.5.1 The General Security Solutions

Concepts analyzed from literature reviews leads us to propose an architecture for our proposed framework that realizes a secured channel of transaction for strengthening the security of mobile cloud computing using hybrid authentication**.** The proposed system architecture is based on the mobile cloud computing (MCC) architecture with some additional security algorithms, protocols and cloud modeling.

3.5.2 The Proposed Security Mechanism

The proposed solutions provide a mechanism for securing end-to-end mobile cloud computing transactions. For our solutions, we proposed the AES symmetric encryption algorithm is used for encryption purposes. Here we plan to make more secure the user's username and password during transportation by using an AES encryption algorithm because the attackers can't access the user's username and password even if they get the username and password, then the second-level authentication can protect against such attacks. To start using cloud services first, the user registers his/her username/password with the fingerprint to respected cloud service providers. The CS provider verifies the details of the customer and saves the username/password with the fingerprint in the authentication server database.

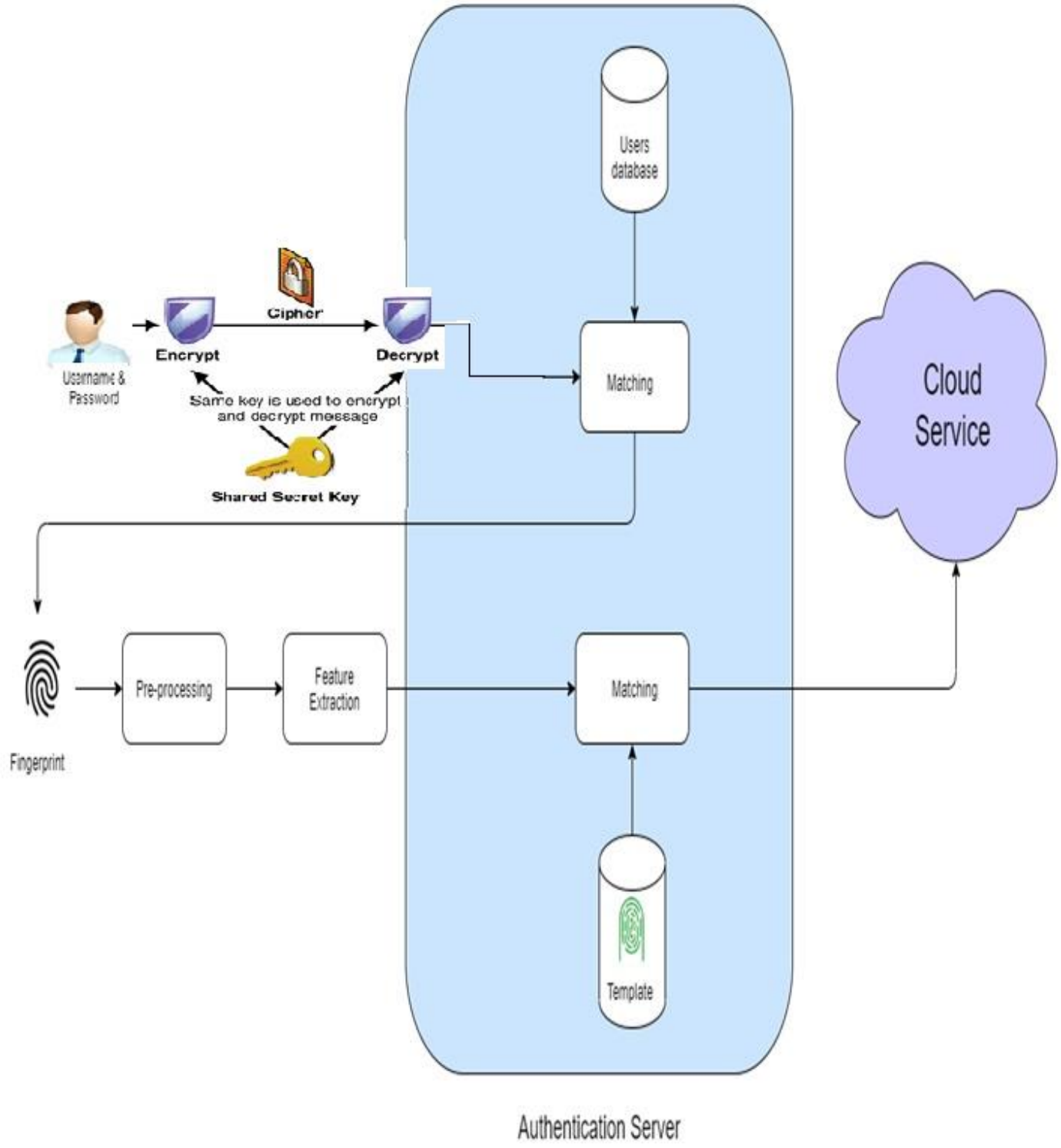

Figure: 3. 3 Proposed Client-Server communications

The communication between the client and the server is encrypted. In this encryption we use the Advanced Encryption Standard (AES) symmetric encryption algorithm. When the client request to access the CS it's expected to enter their username and password the server decrypts the client's username/password and check the user id if it's correct then the client requested to scan his/her fingerprint for the next level of security again if it is correct the user can access the cloud services.

Algorithm

Step 1: Check the entered username, if the user is available in the authentication server or not. If it is available then go to step2 otherwise go to step 6

Step 2: Take the suitable password (decrypted) from the authentication server which is equivalent to entered username, then go to step 3 otherwise go to step 6

Step 3: count the trial, if the trial greater than 3 go to step 4

Step 4: If the count is equal to one or two trial and successfully login then proceed for the next security level, if Count is greater than 3 blocks the username and notify and go to step 6

Step 5: asks the user to enter his/her fingerprint from his/her smart phone fingerprint button, if it matches then access the cloud service otherwise go to step 7

Step 6: Display the error message the "username or password you entered is not correct"

Step 7: the fingerprint you entered is not matched with the template stored in the server, and then terminate the access

Step 9: If the user forget the password then go to forget password module

Step 10: Get email ID as input then send the auto generated verification code to email address

3.5.3 End-to-end Security

The best and most advantageous solution is to deploy the end-to-end security or security at the application layer. Most CS security vulnerabilities do not aim ordinary people, and their targets are usually restricted to special groups so it is reasonable and economical that such groups make their communications secure using end-to-end security mechanisms. Since the encryption and security establishment is performed at the end entities, any change to the MCC hardware not required. In this way, even if the conversation is eavesdropped by the police or legal organizations, they cannot decrypt the transmitted data without having the true ciphering key. Therefore, in order to avoid illegal activities, it should be transparent to the service provider. It may also be necessary to find solutions for a legal interception or a key screw scheme. The end-to-end security establishment has a complete flexibility to the deployed algorithms so the appropriate upgrades can be easily accomplished when necessary.

3.5.4 AES Algorithm

Advanced encryption standard (AES) algorithm is an encryption algorithm to maintain data confidentiality. Both hardware and software are implementations are faster still it is the new encryption standard recommended by NIST (national institute of standards and technology) to replace the digital signature based algorithm named DES[55]. According to[56], AES is more secure and efficient compared to DES and DES3. Summary of research results compared the performance of AES, DES and DES3. Another research result presented in [57] stated why AES is preferable over others is stated as follows:

- AES performs consistently well in both hardware and software platforms under wide range of environments this include 8-bit and 64-bit platforms.
- Its inherent parallelism facilitates efficient use of processor resources resulting in very good software performance
- It requires less memory for implementation making it suitable for restricted space environments memory constrained system like mobile phones.
- The structure has good potential for benefiting from instruction level parallelism
- There are no serious week keys in AES.
- It supports any block size and key sizes that are multiples of 32.

- Statistical analysis of the cipher text has not been possible even after using a huge number of taste case
- No differential and linier cryptanalysis attack has been yet approved on AES.

A performance comparison among AES, DES and triple DES for different micro controllers shows that AES has computational cost of the same order as required for triple DES [57].

Another performance evaluation reveals that AES has advantage over other algorithms like 3DES, DES and Ron's code2 (RC2) in terms of execution time with different packet sizes and throughput for encryption as well as decryption [58]. Moreover, in the case of changing data types such as image instead of text it has been found that AES has an advantage over RC2, Rivest cipher6 (RC6) and blowfish in terms of computational time consumption [57].

CHAPTER FOUR

IMPLEMENTATION AND ANALYSIS OF THE PROPOSED SYSTEM

4.1 Overview of the section

Implementation is the understanding of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy. In other words, an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through programming and deployment.

In this chapter, we include the mobile cloud computing prototypes of the mobile and cloud environment security, result, and analysis of the proposed framework. In this study we use the IDE android studio, in the back end we use firebase, and we also use a java programming language to strengthen the authentication in mobile cloud computing.

4.2 Modeling of the proposed Solutions

In this research we have designed a new efficient and secured model, to see the efficiency and security of the proposed hybrid authentication; we develop a prototype by using IDE android studio with the back end of firebase and tested it. In this model first, the user enters their username and password, if the username and password that entered are correct or match with the stored on the authenticated server then the user requested to scans the user's fingerprints which are considered a new and good idea because it provides more security to the mobile cloud computing. In this model, the user must enter his/her username and password with a fingerprint to authenticate himself on the mobile cloud computing authentication server to access the cloud services where the suggested solution goes by saving the username/password and fingerprints for the authorized users.

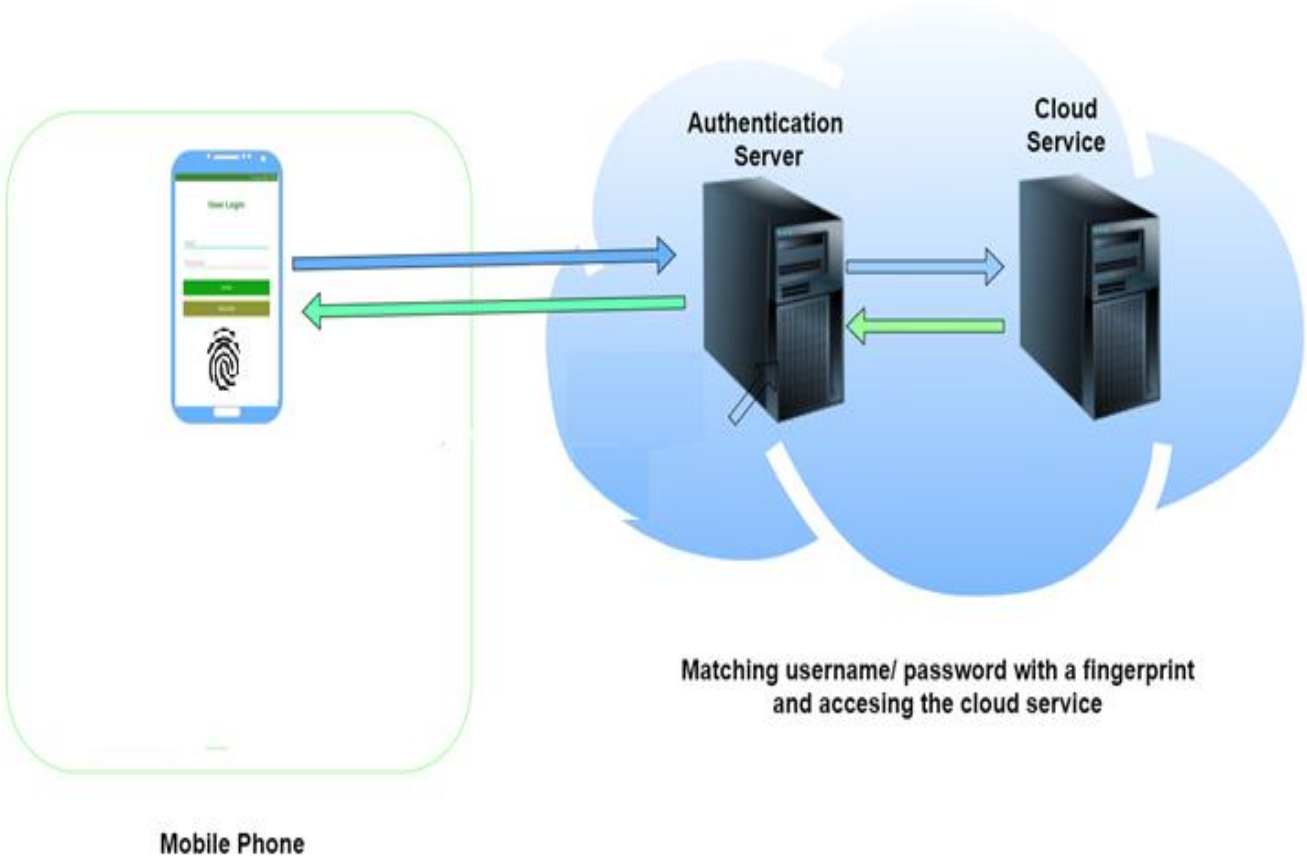

Figure: 4 1 username/password with a fingerprint authentication model

The problem of using username and password is very exposed to the attackers and most of the services may be attacked by the attackers. To overcome such security issues, we proposed new solutions by using combining the username/password with a fingerprint on the authentication. Using two-factor authentication can secure the mobile cloud computing environments. When the user wants to access the account, he/she must enter the defined username/password as the proposed model, the entered username/password will match with the store one, if it matches then the fingerprint will enter to authenticate the user correctly. If both are matched, then the user is authorized to access the cloud services and can get benefit from the cloud computing utilities; Figure 4.1 shows the proposed model. In this research, we use the basic username/password,

fingerprint feature extraction, and the last is for the matching process. Our system consists of two stages the first one is to store the username/password with fingerprints in the cloud authentication server's database and the second stage matching the username/password with fingerprints that were previously stored in the cloud authentication server's database.

4.2.1 Matching the username/password and fingerprints

In this phase the user must enter his/her username/password if it is correct, then authentication server requested the user to enter the next level of security which is scanning the user fingerprint and if it is valid, then the user can access the cloud service and if the scanned fingerprint is wrong the user rejected because the user is unauthorized. After the user fingerprint has been read and send to the database, the system match it with the fingerprint that was previously stored in the cloud authentication server's database and if the matching is correct, the user is considered as authorized and any process can be done on the cloud account. User at this stage can be either authorized or unauthorized.

4.2.2 Authorized User

The user in this step can choose his/her username/password with fingerprint, so that he/she can verify his/her identity on the mobile cloud computing, after that the entered user's fingerprint is matched with his/her previously saved fingerprints. And if it is matched successfully, the user can able to access the mobile cloud computing services.

4.2.3 Unauthorized User

The unauthorized user is a person who doesn't have the correct (actual) credentials to access the services, and if he succeeds in entering with the username/password correctly, he still have to enter his fingerprint. The system matches the user's entered fingerprint with the previously saved fingerprints of the user. Then the authentication server prevents him from accessing the mobile cloud computing because of the lack of matching between the entered fingerprint and the stored ones.

4.2.4 Tools and Technologies Used

The proposed system is a new authentication platform for mobile cloud computing. Here, two emerging trends are used that are cloud and mobile, these two trends are combined and challenging the security performance.

The proposed mobile cloud computing secured the security system through username/password with a (fingerprint) biometric authentication. In this process, the user registers her/his details such as username, password and fingerprint and logins the system with that specified password and fingerprint. Figure: 4.2 give the username and password if the value is correct then he/she goes to next level of security which is entering the user fingerprint. The request is communicated by the client to access the cloud service is shown in Figure: 4.4.

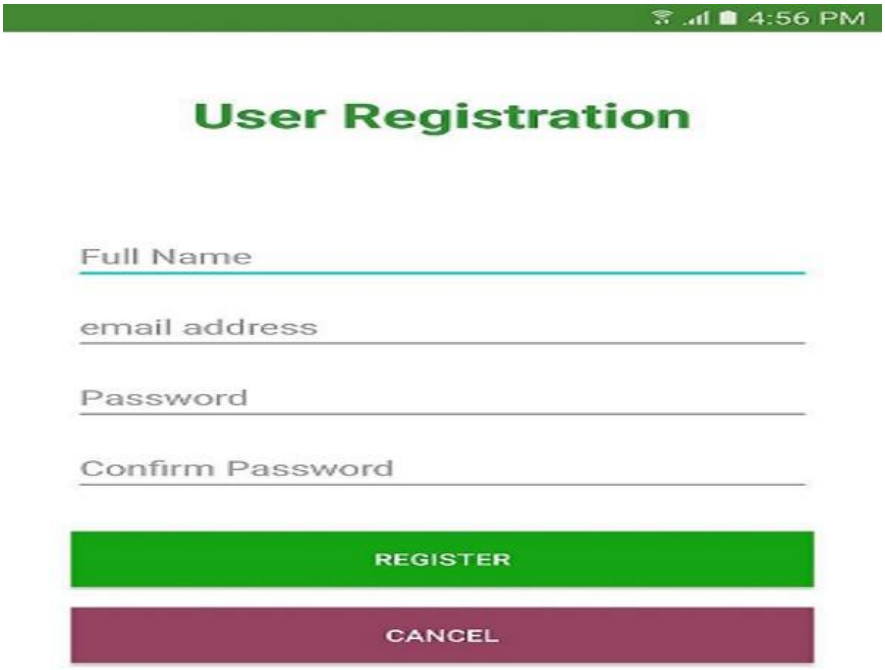

Figure: 4 2 User registration form

```
1. Registration
mAuth.createUserWithEmailAndPassword(email, password)
        .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                if (task.isSuccessful()) {
                    // Sign in success, update UI with the signed-in user's information
                    //FirebaseUser user = mAuth.getCurrentUser();
                    User user = new User(fullName, email);
                    FirebaseDatabase.getInstance().getReference("Users")
                            .child(FirebaseAuth.getInstance().getCurrentUser().getUid())
                            .setValue(user).addOnCompleteListener(new OnCompleteListener<Void>() {
                        @Override
                        public void onComplete(@NonNull Task<Void> task) {

                            if(task.isSuccessful()){
                                //Show Success Message and Go to home

                            }else {
                                                                    ...
                                // User Create Failed
                            }
                        }
                    });

                } else {
                                    ...
                    // Show Error Message
                }
            });
```

Figure: 4 3 source code for user registration

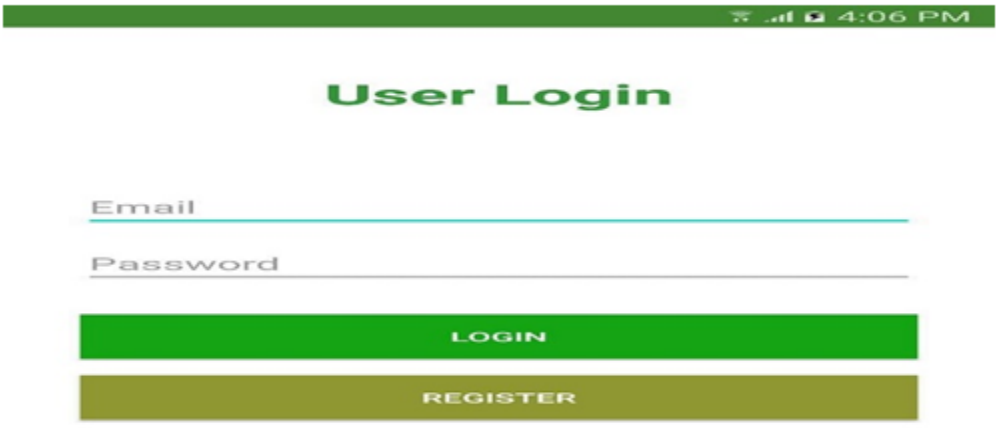

Figure: 4 4 first login by using username and password

2. Sign in username and password

```
mAuth.signInWithEmailAndPassword(email, password)
        .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
           @Override
           public void onComplete(@NonNull Task<AuthResult> task) {
              if (task.isSuccessful()) {
                 FirebaseUser firebaseUser = FirebaseAuth.getInstance().getCurrentUser();
                 if(firebaseUser!=null) {
                    user = new User(firebaseUser.getEmail(), firebaseUser.getDisplayName());
                    user.setEmail(firebaseUser.getEmail());
                    user.setFullName(firebaseUser.getDisplayName());
                 }

                 authenticateUsingFingerprint();
              } else {
                 ...
                                                    //Show Error Dialog
                   }

              }
           }
        });
}
```

Figure: 4 5 source code for user login

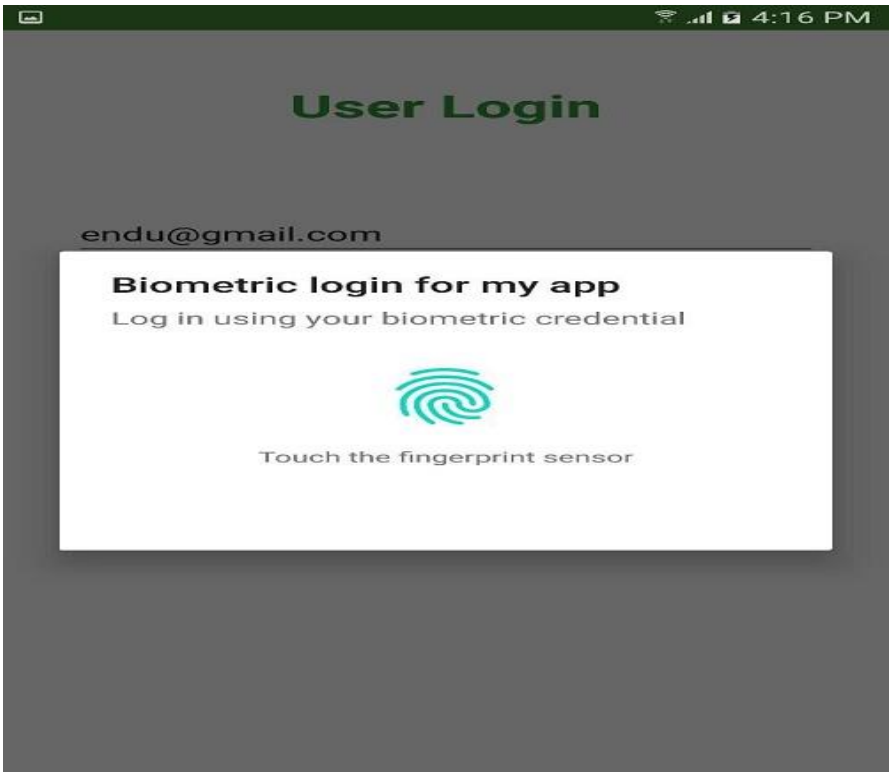

Figure: 4 6 biometric login

```
3. BioMetric Authentication

BiometricPrompt biometricPrompt = new BiometricPrompt(MainActivity.this,
            executor, new BiometricPrompt.AuthenticationCallback() {
        @Override
        public void onAuthenticationError(int errorCode,
                                          @NonNull CharSequence errString) {
            super.onAuthenticationError(errorCode, errString);
        }

        @Override
        public void onAuthenticationSucceeded(
                @NonNull BiometricPrompt.AuthenticationResult result) {
            super.onAuthenticationSucceeded(result);
            Intent homeIntent = new Intent(MainActivity.this, Home.class);
            homeIntent.putExtra("user_extra", user);
            startActivity(homeIntent);
        }

        @Override
        public void onAuthenticationFailed() {
            ...
                        //Authentication Failed Message will be displayed.
        }
    });

BiometricPrompt.PromptInfo promptInfo = new BiometricPrompt.PromptInfo.Builder()
            .setDeviceCredentialAllowed(false)
            .build();
    biometricPrompt.authenticate(promptInfo);
```

Figure: 4 7 source code for user biometric login

We used the following tools for the developments:

IDE - Android Studio**: -** An IDE or Integrated Development Environment is a comprehensive solution that gives mobile app developers the opportunity to perform the software development cycle continuously and at a faster pace. The development cycle includes designing, writing, compiling, testing, and debugging the code.

At the back end of the development we used a Firebase. Firebase is a mobile platform that helps you quickly develop high-quality apps, grow your user base, and earn more money. Firebase is made up of complementary features that you can mix-and-match to fit your needs, with Google Analytics for Firebase at the core. We use a java Programming Language to develop the proposed system. Java is a powerful general-purpose programming language. It is used to develop desktop and mobile applications, big data processing, embedded systems, and so on.

4.3 Evaluation of the Proposed Solutions

4.3.1 Attack Effort of Common Evaluation Methodology (CEM)

Attack effort refers to a function of skill, resource, and motivation presented by Criteria Evaluation Methodology (CEM) in CC, which is based on time elapsed, expertise, knowledge of the target of the attack, easy exposure to the attack and equipment. Attack effort quantifies the attack effort of the target of attack by assigning values to each element.

4.3.2 Elapsed time

Elapsed time is the total amount of time taken by an attacker to identify that a particular effort vulnerability, to develop an attack method and to sustain effort required to perform the attack against the TOE.

4.3.3 Expertise

Expertise refers to the level of generic knowledge of the underlying principles, product type or attack methods. The identified levels are as follows:

• ***Layman***: unknowledgeable compared to experts or proficient persons, with no particular expertise

• ***Proficient***: knowledgeable in that they are familiar with the password attack tools and methods

• ***Expert***: familiar with implementing in password attack tools, operation algorithm of password authentication systems.

4.3.4 Knowledge of target of attack

Knowledge of the target of evaluation (TOE) refers to specific expertise in relation to the TOE. This is distinct from generic expertise, but not unrelated to it. Identified levels are as follows:

• ***Public***: information gained from the Internet

• ***Restricted***: knowledge that is controlled within the developer organization and shared with other organizations under a non-disclosure agreement

• ***Sensitive***: knowledge that is shared between discreet teams within the developer organization, access to which is constrained only to members of the specified teams

• ***Critical***: knowledge that is known by only a few individuals, access to which is very tightly controlled on a strict need to know basis and individual undertaking

4.3.5 Period of easy exposure to attack

Period (chance) related to elapsed time, when an attacker can approach the target of attack.

• ***Unnecessary/unlimited access:*** the attack doesn't need any kind of opportunity to be realized because there is no risk of being detected during access to the TOE.

• ***Easy***: access is required for less than a day

• ***Moderate***: access is required for less than a month

• ***Difficult***: access is required for at least a month

4.3.6 Equipment

Equipment refers to the equipment required to identify or exploit vulnerability.

• ***Standard*** equipment is readily available to the attacker, either for the identification of a vulnerability or for an attack.

• ***Specialized*** equipment is not readily available to the attacker, but could be acquired without undue effort.

• ***Bespoke*** equipment is not readily available to the public as it may need to be specially produced.

• ***Multiple Bespoke*** is introduced to allow for a situation, where different types of bespoke equipment are required for distinct steps of an attack.

Table 4 1 (based on [59]) identifies the factors and associates numeric values with each level.

Factor	Level	Value
Elapsed time	≤ 1 day	0
	≤ 1 week	1
	≤ 1 month	4
	≤ 3 months	10
	≤ 6 months	17
	> 6 months	19
	not practical	∞
Expertise	Layman	0
	Proficient	3
	Expert	6
	Multiple experts	8
Knowledge of TOE	Public	0
	Restricted	3
	Sensitive	7
	Critical	11
Window of opportunity	Unnecessary/unlimited	0
	Easy	1
	Moderate	4
	Difficult	10
	None	∞
Equipment	Standard	0
	Specialized	4
	Bespoke	8
	Multiple bespoke	9

To determine the attack effort for an attack, we sum up the appropriate values for the factors from Table 4.1 and apply Table 4.2 to map the sum to the attack effort.

Table 4 2 Rating of attack effort

Values	Attack effort required to identify and exploit vulnerability
0–9	Basic
10–13	Enhanced-Basic
14–19	Moderate
20–24	High
≥ 25	Beyond High

4.4 Most common attack on fingerprint authentication

4.4.1 Attack effort for lifting latent fingerprints from surface

Unlike PINs (Personal Identification Numbers) and passwords, fingerprints are not secret. They can be lifted from surfaces touched by the victim of the attack. Once a usable image has been obtained, fingerprint dummies may be fabricated.

Table 4 3 Attack effort estimate for lifting a latent fingerprint from a touched surface

Attack	Elapsed time	expertise	Knowledge of TOE	Window of opportunity	equipment	Required attack effort	
						sum	Rating
Lift a latent fingerprint from a touched surface	0	6	0	10	4	20	High

4.4.2 Fingerprint attack effort for fabricating dummies

Fingerprint dummies for spoofing fingerprint recognition system can be fabricated from different materials. Depending on the sensor technology, the dummy need to imitate certain physical characteristics of fingers measured by fingerprint sensor in order to allow an attacker to successfully spoof the sensor

Table 4 4 Attack effort estimate for fabricating fingerprint dummies

Attack	Elapsed time	expertise	Knowledge of TOE	Window of opportunity	equipment	Required attack effort	
						Sum	Rating
Fabricate a dummy from a given fingerprint image	1	8	0	0	4	12	Enhanced-Basic

4.5 Most common attack effort on the username/password authentication

4.5.1 Brute-force attack

Brute force attack is one of the most common forms of password attack method and the easiest for hackers to perform. In a brute force attack, a hacker uses a computer program to log into a user's account with all possible password combinations. Moreover, brute force accounts don't start at random; instead, they start with the passwords that are easier to guess. In cryptography, a brute force attack consists of an attacker submitting numerous passwords or passphrases in the hope of possibly guessing a combination correctly. The attacker systematically checks all possible passwords and passphrases until the correct one is found. Alternatively, the attacker can attempt to guess the key which is usually created from the password using a key derivation function. This is known as an exhaustive key search.

Table 4 5 Attack effort estimate for brute-force attack

Attack	Elapsed time	expertise	Knowledge of TOE	Window of opportunity	equipment	Required attack effort	
						sum	Rating
Brute force	1	6	0	10	0	17	Moderate

4.5.2 Key logger attack

It is easy to learn how to use a keylogger. Once you have installed a keylogger on the phone you want to monitor, it will run invisibly in the background and detect all activities that take place on that device. The keylogger software will record all the keys pressed and also record the conversations written in the chat. The downside of a keylogger is that it can be used by hackers for illegal purposes, but it also has a positive side. Invisible Keylogger is the perfect tool for parents who want to monitor conversations and know what they are doing online.

Table 4 6 Attack effort estimate for lifting a latent fingerprint from a touched surface

Attack	Elapsed time	expertise	Knowledge of TOE	Window of opportunity	equipment	Required attack effort	
						sum	Rating
Key logger	0	6	0	0	4	10	Enhanced-Basic

Table 4 7 Attack effort estimate for hybrid authentication technique

Attack	sum	Rating
Lift a latent fingerprint from a touched surface with brute force attack	37	**Beyond high**
Lift a latent fingerprint from a touched surface with keylogger attack	30	**Beyond high**
Fabricate a dummy from a given fingerprint image with brute force attack	29	**Beyond high**
Fabricate a dummy from a given fingerprint image with keylogger attack	22	**high**

4.6 Discussion on the result

To make the evaluation, first we select two most common attacks from the username/password which is a brute force attack and a keylogger attack and from the finger print perspective we also select two most common attacks which is a lift a latent fingerprint from a touched surface and a fabricate a dummy from a given fingerprint image attacks. In the table 4.2, we categorize the rating of attack effort which requires identifying and exploiting vulnerabilities and we give the score points professionally by referring table 4.1 [59] for each attack types. Then by adding the values we give the required attack effort rates for each of them. Finally, by using the attack effort rates we combined these attack types to evaluate our propose solutions. As table 4.6 indicates, the combination of most common attacks on MCC has an attack effort required to identify and exploit vulnerability value of *High* or *Beyond High*. High attack effort required to identify and exploit vulnerability means sophisticated technique and skill is required to bypass the system and low attack effort required means that the system can easily get bypassed. Therefore, this evaluation proves that our proposed hybrid authentication technique requires more effort to get bypassed.

Comparison of existing approaches vs. the proposed approach

Table 4 8 comparison between existing vs. proposed approaches

Existing approaches	Proposed Approach
Username and password: is more exposed to the guessing and brute force attack and also the user may forget their password	Using a combination authentication is provides more security than the traditional one. Implemented additional security enhancement mechanism after username/password.
Smart card with PIN: if a user loses his/her smart card, then he/she cannot authenticate and access the CS and also exposed for bypass attack.	There is need to hold any additional equipment or devices simply the user uses his/her username and fingerprint only
Fingerprint: fake fingerprint may prepare by making artificial fingerprint out of plasters and silicon.	Here in our proposed system if the attacker passes the username/password then the fingerprint can protect and in our system we set only three incorrect trial if the attacker tries more than three times the system automatically blocks the attacker
Password and Keystroke: if the user writing speed changed and /or change the hardware may affect the writing speed then the authentication is not allow to access the CS	The MCC users need only his/her password and scanning his/her fingerprint

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORKS

5.1 Conclusions

The general objective of this research is to enhance user authentication in mobile cloud computing and propose a new authentication security architecture to enhance security and privacy in mobile cloud computing. To understand this study, assessment of recent work on the area of data security related to mobile cloud computing, authentication security issues and solutions, reference architectures and related studies have been reviewed. From the assessments, we deserve to design and implemented a secure authentication framework which addresses the current security holes related to the authentication attacks. Moreover, we implement a secure authentication to maximize the communication security for the mobile cloud computing services.

Mobile Cloud computing has many issues and one of the major issues is security as the user is storing his/her personal information outside the mobile device, making financial transactions and other important information are stored in the Mobile Cloud environment. To protect the information or data, a proper authentication is necessary to ensure that only authorized user has access information stored in the cloud. It has been found that some existing biometric techniques can provide effective authentication in a Mobile Cloud computing environment. After studying a number of existing biometric techniques, a new authentication technique based on the combination of username/password with biometrics (fingerprint) has been proposed. This technique uses a username/password and a built-in mobile fingerprint scanner. If the user wants to access the cloud from mobile, he/she needs to give email id as a username with password and for the next level security he/she use a fingerprint button from his/her smart mobile, both the username/password and a biometric (fingerprint) is used for the purposes of authentication. The username/password communication is encrypted by using the advanced encryption standard to make the security reliable and for the biometric (fingerprint) the Features are extracted from the scanned fingerprint, and then store the templates. When the user wants to login to the cloud service, the user username/password and the fingerprint compare with the stored username/password and the fingerprint templates respectively. The proposed username/password with a biometric (fingerprint) authentication technique gives better

performance than existing techniques in the mobile cloud computing. For effective user authentication in the cloud computing environment, the authentication technologies described above should be used by combining them appropriately or a secure user authentication method should be developed for the right purpose of mobile cloud computing services.

So, implementing the proposed strengthening of the authentication framework to the mobile cloud computing system, the mcc authentication would be smarter and secured compared to the existing frameworks and it is also a best practice for the resource constrained devices.

5.2 Future Works

Although, the proposed authentication solution solves some issues there are some issues unsolved yet, regarding the authentication and data security as a whole on MCC environment that deserves further study and analysis with better algorithms and protocols. In the future scholars can add the authentication security by using a security algorithms, it's open for the fingerprint authentication securities.

Note that, the study should also take into consideration the fact that the MCC is emerging and constantly changing computing so, the cryptosystems, protocols and access control mechanisms are working with the risks identified within this study and may need to be updated accordingly in future.

REFERENCES

[1] A. N. Khan, M. M. Kiah, S. U. Khan, and S. A. Madani, “Towards secure mobile cloud computing: A survey,” *Future Generation Computer Systems*, vol. 29, no. 5, pp. 1278 1299, 2013.

[2] M. Sarvabhatla and C.S. Vorugunti, "A robust mutual authentication scheme for data security in cloud architecture," in Communication Systems and Networks (COMSNETS), 2015 7th International Conference on, Jan 2015, pp. 1-6.

[3] S. Ruj, M. Stojmenovic, and A. Nayak, "Decentralized Access Control with Anonymous Authentication of Data Stored in Clouds," Parallel and Distributed Systems, IEEE Transactions on, vol. 25, no. 2, pp. 384-394, Feb 2014.

[4] S. Grzonkowski, P.M. Corcoran, and T. Coughlin, "Security analysis of authentication protocols for next-generation mobile and CE cloud services," in Consumer Electronics - Berlin (ICCE-Berlin), 2011 IEEE International Conference on, 2011, pp. 83-87.

[5] R.K. Lomotey and R. Deters, "SaaS Authentication Middleware for Mobile Consumers of IaaS Cloud," in Services (SERVICES), 2013 IEEE Ninth World Congress on, 2013, pp. 448-455.

[6] A. J. Aviv, K. Gibson, E. Mossop, M. Blaze, and J. M. Smith, “Smudge attacks on smartphone touch screens,” in Proceedings of the 4th USENIX Conference on Offensive Technologies, WOOT’10, pp. 1–7, 2010.

[7] K. D. Mitnick and W. L. Simon. The Art of Deception: Controlling the Human Element of Security. John Wiley & Sons, Inc., New York, NY, USA, 2003. ISBN 076454280X.

[8] D. V. Klein. ”foiling the cracker”: A survey of, and improvements to, password security, 1990.

[9] Z. Xu, K. Bai, and S. Zhu, “TapLogger: Inferring user inputs on smartphone touchscreens using on board motion sensors,” in Proceedings of the Fifth ACM Conference on Security and Privacy in Wireless and Mobile Networks, WISEC ’12, pp. 113–124, 2012.

[10] H. Khan, A. Atwater, and U. Hengartner, “Itus: An implicit authentication framework for Android,” in Proceedings of the 20th ACM Annual International Conference on Mobile Computing and Networking, MobiCom ’14, pp. 507–518, 2014.

[11] S. Prabhakar, S. Pankanti, and A. K. Jain, “Biometric recognition: security and privacy concerns,” IEEE Security Privacy, vol. 99, no. 2, pp. 33–42, Mar. 2003.

[12] P. Campisi, E. Maiorana, M. Lo Bosco, and A. Neri, “User authentication using keystroke dynamics for cellular phones,” IET Signal Processing, vol. 3, no. 4, pp. 333–341, 2009.

[13]. Z. Sanaei, S. Abolfazli, A. Gani, and R. Buyya, Heterogeneity in mobile cloud computing: Taxonomy and open challenges, IEEE Communication Surveys and Tutorials, 16(1), 369–392, 2014.

[14]. N. Fernando, S. W. Loke, and W. Rahayu, Mobile cloud computing: A survey. Future Generation Computer Systems, 29(1), 84–106, 2013.

[15]. H. T. Dinh, C. Lee, D. Niyato, and P. Wang, A survey of mobile cloud computing: Architecture, applications, and approaches, Wireless Communications and Mobile Computing, 13(18), 1587–1611, 2013.”

[16] Armbrust M, Fox A, Griffith R, et. al. Above the Clouds: A Berkeley View of Cloud Computing Michael. *Technical Report* UCB/EECS- 2009-28, Berkeley, 2009.

[17] Abolfazli S, Sanaei Z, Ahmed E, Gani A and Buyya R. Cloud-Based Augmentation for Mobile Devices: Motivation, Taxonomies, and Open Challenges. In *IEEE Communications Surveys & Tutorials* 2014;16(1): pp. 337-68.

[18] Mell P and Grance T. The NIST Definition of Cloud Computing. *NIST Special Publication* 800-145, 2011.

[19] Sanaei Z, Abolfazli S, Gani A, Shiraz M. SAMI: Service-based arbitrated multi-tier infrastructure for Mobile Cloud Computing. In *IEEE Intl conf. Communications in China (ICCC)*, pp.14,19, 15-17 Aug. 2012

[20] Dijiang H, Zhibin Z, Le Xu, Tianyi X, Yunji Z. Secure data processing framework for mobile cloud computing. *INFOCOM WKSHPS*. pp.614- 618, 10-15 April 2011.

[21] Bhattacherjee S, Majumder S, De D. Trust model for femto-cloud based mobile network. In *5th International Conference on The Next Generation Information Technology Summit (Confluence)*, pp.53,58, 25-26 Sept. 2014.

[22] Zhang JY, Pang Wu, Jiang Zhu, Hao Hu, Bonomi F. Privacy-Preserved Mobile Sensing through Hybrid Cloud Trust Framework. *In Sixth IEEE International Conference on Cloud Computing (CLOUD)*, pp.952-953, June -July 2013.

[23] Dinh, H.T., Lee, C., Niyato, D., Wang, P., 2011. A survey of mobile cloud computing: architecture, applications, and approaches. Wireless Commun. Mobile Comput. 13 (18), 1587 1611.

[24] Chetan, S., Kumar, G., Dinesh, K., Mathew, K., Abhimanyu, M., 2010. Cloud Computing for Mobile World. Accessed: 27/03/2017, Available at: http://chetan.ueuo.com/projects/CCMW.pdf.

[25] Lee. Chonho, 2014. a sruvey of mobile cloud computing: architecutre, applications and approaches. Wireless communications and mobile computing, 1(1), pp. 1-38.

[26] S. William, Cryptography and Network Security Principles and Practice Book, 5th ed.

[27] J. Z. Ansah and M. Mehreen, "Encryption/Decryption Using Elliptic Curve Cryptography," *International Journal of Advanced Research in Computer Science*, vol. 8, July - August 2017.

[28] D. Feng,M. Zhang,Y. Zhang and Z. Xu, "Study on cloud computing security," Journal of Software ,vol. 22,no. 1,2011,pp. 71- 83.

[29] M. Al Morsy, J. Grundy and I. Müller, "An Analysis of The Cloud Computing Security Problem", In Proceedings of APSEC 2010 Cloud Workshop, Sydney, Australia, (2010), November 30.

[30] Soeung-Kon, J. -H. Lee and S. W. Kim, "Mobile Cloud Computing Security Considerations", Journal of Security Engineering, no. 9, (2012) April.

[31] K. opovi and Z. Hocenski, "Cloud computing security issues and challenges", MIPRO, 2010 Proceedings of the 33rd International Convention, (2010) May 24-28.

[32] D. Chen and H. Zhao, "Data Security and Privacy Protection Issues in Cloud Computing", International Conference on Computer Science and Electronics Engineering (ICCSEE), (2012) March 23-25.

[33] Morshed, M. S. Jahan, M. M. Islam, M. K. Huq, M. S. Hossain and M. A. Basher, "Integration of Wireless Hand-Held Devices with the Cloud Architecture: Security and Privacy Issues", International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), (2011) October.

[34] Locations of Google's Data Centers:⟨http://www.google.com/about/ datacenters/inside/locations/index.html⟩

[35] M. Esmaili, R. Safavi-Naini, Y. Zheng: Authentication Techniques: The Center for Computer Security Research.

[36] Chang-Lung Tsai and Uei-Chin Lin, (February 2011) Information Security of Cloud Computing for Enterprises, Volume 3, issue1.16, Number 1.

[37] Markus Johnsson & A.S.M FaruqueAzam,(March 2011) Mobile One Time Passwords and RC4 Encryption for Cloud Computing, Master Thesis,Computer and Electrical Engineering Halmstad University.

[38] X. Wang,M. Chen,T. Kwon,L. Yang and V. Leung,"AMES-Cloud: ramework of adaptive mobile video streaming and efficient social video sharing in the clouds," IEEE Transactions on Multimedia, 10.1109/TMM.2013.2239630,Feb. 2013.

[39] M. Sarvabhatla and C.S. Vorugunti, "A robust mutual authentication scheme for data security in cloud architecture," in Communication Systems and Networks (COMSNETS), 2015 7th International Conference on, Jan 2015, pp. 1-6.

[40] F. Omri, R. Hamila, S. Foufou, and M. Jarraya, "Cloud-Ready Biometric System for Mobile Security Access," Networked Digital Technologies, pp. 192-200, 2012.

[41] I. Al Rassan and H. AlShaher, "Securing Mobile Cloud Computing Using Biometric Authentication (SMCBA)," 2014 Int. Conf. Comput. Sci. Comput. Intell., pp. 157–161, 2014.

[42] H.Dinesha and V. Agrawal, "Multi-level authentication technique for accessing cloud services," in Computing, Communication and *Applications (ICCCA), 2012 International Conference on*, 2012, pp. 1-4.

[43] D. S. Oh, B. H. Kim, and J. K. Lee, "A Study on Authentication System Using QR Code for Mobile Cloud Computing Environment," *Future Information Technology*, pp. 500-507, 2011.

[44] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless Communications and Mobile Computing*, 2011.

[45] A. (2014). A multifactor biometric authentication for the cloud. Computational Intelligence, Cyber Security and Computational Models, Advances in Intelligent Systems and Computing (Vol. 246, 395-403).

[46] An efficient remote user authentication and key agreement protocol for mobile client-server environment from pairings. Ad Hoc Networks. 2012;10(6):1009–16.

[47] Jin H, Ibrahim A, Qiang W, Zou D. Efficient password-based two factors authentication in cloud computing. Int J Secur its Appl. 2012;6(2):143–148.

[48] "A Secure and efficient Biometric authentication as a service for cloud computing," IEEE, October 09-11 2014

[49] K. K. Kavitha, B. L. Gopinath, C. U. Kushalappa, and D. K. H, “Mobile Cloud Computing With A Private Authentication Scheme,” pp. 172–176, 2016

[50] W. T. Meshach and K. S. S. Babu, “Secured and Efficient Authentication Scheme for Mobile Cloud (2013),” vol. 2, no. 1, pp. 242–248, 2013

[51] M. A Lightweight Authentication Scheme for Mobile Cloud Computing. Int. J. Comp. Sci. Bus. Inf. 2014, 14, 153–160.

[52] Técnico, I.S.; Telecomunicações, I.D.; Correia, P.L. Smartphone-based palmprint recognition system. In Proceedings of the 2014 21st International Conference on Telecommunications (ICT), Lisbon, Portugal, 4–7 May 2014; pp. 457–461.

[53] D.; Yang, L. Entity authentication in a mobile-cloud environment. In Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop, Oak Ridge, TN, USA, 8–10 January 2013; pp. 1–4.

[54] Tasia, T.-Y. Chang, P.-C. Cheng and J.-H. Lin, “Two novel biometric features in keystroke dynamics authentication systems for touch screen devices,” Security and Communication Networks, vol. 7, no. 4, pp. 750–758, Apr. 2014.

[55] KawserWazedNafi1 and Dr.M.M.AHashem,"anewer user authentication ,file encryption and distributed server based architecture," 2013.

[56] Mahajan,prerna and A.sachdeva "study of encryption algorithm AES,DES,RSA for security," in computer science and technology, 2013.

[57] B.Hamdan.O.Alanazi, "new comparative study between DES,3DES,and AES within factors," journal of computing, vol. 2, 2010.

[58] Punithasurya and S.jebapriya, "analisis of different access control mechanism in cloud," international journal of applied information systems, 2012.

[59] International Standard ISO/IEC 18045. Information technology – Security techniques – Methodology for IT security evaluation

[60] J. A. O. D. #. H. F. #. Moh'd Fawzi Al-Hunaity #1, "Security Model for Communication and Exchanging Data in Mobile Cloud Computing," International Journal of Computer Trends and Technology (IJCTT) , vol. 30, pp. 1-2, December 2015.

[61] Xavier Boyen. Hidden credential retrival from a reusable password. In ASI-ACCS '09: proceedings of the 4th International symposium on Information, Computer, and communications security, pages 228-238, New York, NY, USA,2009. ACM.

[62] Dinei Florencio, Cormac Herley, and Baris Coskun. Do strong web password accomplish anything? InHOTSEC;07: proceedings of the 2nd USENIX worship on Hot topics, pages 1-6, CA,USA,2007. USENIX Association.

[63] R. Shirey. Internet Security Glossary, Version 2. RFC 4949 (Information). Internet Engineering Task Force (IETf). http://www.ietf.org/rfc/rfc4949.txt, August 2007.

Annex A: User Registration

```
btnRegister.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
           if(validInput) {
              registerUser(fullName, email, password);
           } } });
public void registerUser(String fullName, String email, String password)
   {
      mAuth.createUserWithEmailAndPassword(email, password)
            .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
               @Override
               public void onComplete(@NonNull Task<AuthResult> task) {
                  if (task.isSuccessful()) {
                     // Sign in success, update UI with the signed-in user's information
                     //FirebaseUser user = mAuth.getCurrentUser();
                     User user = new User(fullName, email);
                     FirebaseDatabase.getInstance().getReference("Users")
                           .child(FirebaseAuth.getInstance().getCurrentUser().getUid())
```

```
                    .setValue(user).addOnCompleteListener(new
OnCompleteListener<Void>() {
                        @Override
                        public void onComplete(@NonNull Task<Void> task) {
                            if(task.isSuccessful()){
                            ….
                            // Registration Successful Message
                                    });
                            }else {
                            ….
                            // Error Message Sent to User
                            }
                        }
                    });
            }
```

Annex B: User Login

```
btnLogin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
           boolean validation = validateInput(email, password);
           if(validation) {
              mAuth.signInWithEmailAndPassword(email, password)
                    .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                       @Override
                       public void onComplete(@NonNull Task<AuthResult> task) {
                          if (task.isSuccessful()) {
                             FirebaseUser firebaseUser =
FirebaseAuth.getInstance().getCurrentUser();
                             if(firebaseUser!=null) {
                                user = new User(firebaseUser.getEmail(),
firebaseUser.getDisplayName());
                                user.setEmail(firebaseUser.getEmail());
                                user.setFullName(firebaseUser.getDisplayName());
                             }
                             authenticateUsingFingerprint();
```

```
                    } else {

                    ….

                    #Show Try Again Message

                            }

                        });

                    }

                }

            });

        }

    }

});
```

Annex C: Biometric login

```
private void authenticateUsingFingerprint() {
    Executor executor = ContextCompat.getMainExecutor(MainActivity.this);
    BiometricPrompt biometricPrompt = new BiometricPrompt(MainActivity.this,
            executor, new BiometricPrompt.AuthenticationCallback() {
        @Override
        public void onAuthenticationSucceeded(
                @NonNull BiometricPrompt.AuthenticationResult result) {
            super.onAuthenticationSucceeded(result);
            Intent homeIntent = new Intent(MainActivity.this, Home.class);
            homeIntent.putExtra("user_extra", user);
            startActivity(homeIntent);}
        @Override
        public void onAuthenticationFailed() {
            super.onAuthenticationFailed();
         // Authentication Failed Error Message
        }  });
    BiometricPrompt.PromptInfo promptInfo = new BiometricPrompt.PromptInfo.Builder()
            .setTitle("Biometric login - AuthPrototype")
```

```
        .setSubtitle("Log in using your biometric credential")
                .setNegativeButtonText(" ")
        .setDeviceCredentialAllowed(false)
        .build();
biometricPrompt.authenticate(promptInfo);}
```