



FINAL REVIEW

RESEARCH ON DIFFERENT USER AUTHENTICATION SYSTEMS AND THEIR LIMITATIONS

GROUP MEMBERS

◆ SHRUTI LAKHARA
20BCY10173

◆ ROHAN
KOLHATKAR
20BCY10177

◆ PRAHASITH
20BCY10142

◆ JOEL FRANKO
20BCY10104

DIFFERENT AUTHENTICATION SYSTEMS

Let's begin.



INTRODUCTION

- In this paper we have included our research on different types of authentication systems and their limitations. We have explained briefly about different types of authentication systems, their disadvantages and their rate of success. We have also included some working protocols and framework of the authentication systems.
- We also performed implementation of 4 different types of authentication systems.
- The most important factor for concerns with the authentication systems is how secure it is and how easy to use it is.
- Also the paper presents some new applications of these authentication systems and models which can be brought into use and is more safe.

EXISTING WORK WITH LIMITATIONS

Cybercriminals always improve their attacks. As a result, security teams are facing plenty of authentication-related challenges. This is why companies are starting to implement more sophisticated incident response strategies, including authentication as part of the process. The list below reviews some common authentication methods used to secure modern systems.

1. Password-based authentication

Passwords can be in the form of a string of letters, numbers, or special characters. To protect yourself you need to create strong passwords that include a combination of all possible options. However, passwords are prone to phishing attacks and bad hygiene that weakens effectiveness.

2. Multi-factor authentication

Multi-Factor Authentication (MFA) is an authentication method that requires two or more independent ways to identify a user. MFA may be a good defence against most account hacks, but it has its own pitfalls. People may lose their phones or SIM cards and not be able to generate an authentication code.

3. Certificate-based authentication

Certificate-based authentication technologies identify users, machines or devices by using digital certificates.

4. Biometric authentication

Biometrics authentication is a security process that relies on the unique biological characteristics of an individual. Despite increased security, efficiency, and convenience, biometric authentication also has disadvantages like False positives and inaccuracy – False rejects and false accepts can still occur preventing select users from accessing systems.

5. Token-based authentication

Token-based authentication technologies enable users to enter their Use cases of token-based authentication include RESTful APIs that are used by multiple frameworks and clients.

PROPOSED WORK AND METHODOLOGY

In this project we did a research on Authentication systems and their limitations. We gathered the data about different authentication systems with reliable sources and then we did a research about some authentication systems and finding their limitations. The research also included a survey on authentication systems. The research paper also contains some new and developing authentication methods. By this research we can get a clear understanding about the authentication Systems and their importance in cyber security. We also implemented four authentication systems namely simple password, fingerprint based, QR based, OTP based.

NOVELTY

So, the main goal here is to know why a particular authentication system is chosen for the particular case and which is the best authentication system .In this project we analyzed the system and the implementations of the systems

- we found that the simple password one is too easy to use. Also, it has less security and can be brute forced.
- The OTP one is bit secured and it can be used as the second factor in two factor authentication.
- The limitation with the biometrics and QR one is that it it requires a device well with has the camera and fingerprint sensor. Also the fingerprint one only works if the user has already setup his fingerprint in a device..

REAL TIME USAGE

- The main usage of research is that we can conclude why a particular authentication system is used here. Generally we don't get deeper into the reason behind this For example IRCTC uses captcha and OTP for account verification, GMAILS uses two-factor authentications nowadays where they send a popup on the main device to confirm login, older Microsoft devices using image-based authentications system when other devices use pattern based and nowadays they shifted to face and fingerprint-based systems.
- The research paper will help in understanding the evolution of authentication system and how step by step we added security to various authentications systems.

HARDWARE AND SOFTWARE REQUIREMENTS

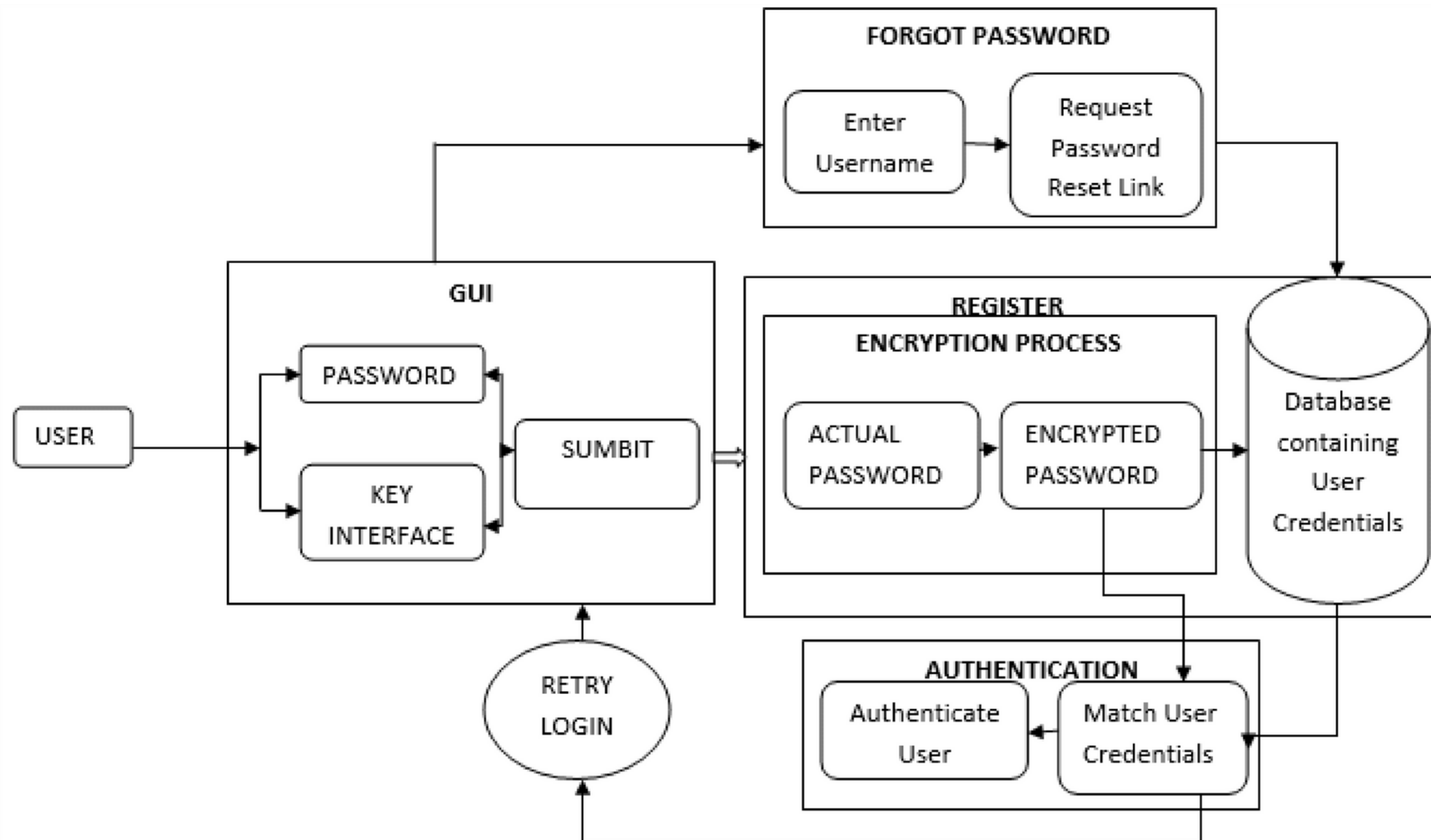
Appliance

Platform	Advanced Authentication Appliance runs 64-bit operating system on x86-64 hardware supported by SLES 12 SP4.
RAM	Minimum: 6 GB Recommended: 12 GB
Processor	Minimum: 4 Cores Recommended: 8 Cores Processor must support SSE 4.2 instructions. For more information about how to check whether the CPU supports SSE 4.2 Instructions on CPU .
Hard Disk space	Minimum: 60 GB Recommended: 100 GB
Virtual System	Supported Virtual systems are: <ul style="list-style-type: none">• Citrix XenServer 7.1, 7.5• Citrix Hypervisor 8.0• Hyper-V Server 2016 or later• VMware ESX 5.5 or later

Windows Client

Processor	Minimum: 2 Cores Recommended: 4 Cores
Hard Disk	Minimum: 100 MB Recommended: 1 GB
Memory	Minimum: 2 GB Recommended: 4 GB
Operating System	Any one of the following operating systems: <ul style="list-style-type: none">• Microsoft Windows 8.1 (32-bit or 64-bit)• Microsoft Windows 10 v1903, v1909, v2004, 20H2 (32-bit and 64-bit)• Microsoft Windows Server 2012 R2• Microsoft Windows Server 2016• Microsoft Windows Server 2019 NOTE: Advanced Authentication Windows Client is not supported on Windows Server Core editions.

OVERALL SYSTEM ARCHITECTURE



TYPES

01

SIMPLE PASSWORD BASED

04

TOKEN BASED

02

GRAPHICAL OR IMAGE BASED

05

QR CODE BASED

03

BIOMETRICS BASED

06

TWO FACTOR AUTHENTICATION

LITRATURE REVIEW

For our research we have used many research papers and websites. And we have explained about each of the research papers detail in our research paper. The research papers and website we have used are.

1. Mushtaq Ali , Amanullah Baloch, Abdul Waheed , Mahdi Zareei , Rimsha Manzoor1, Assam Sajid, And Faisal Alanazi “A Simple And Secure Reformation-based Password Scheme”
2. Nizamani, Shah, Hassan, Raheel, Shaikh, Riaz, Abozinadah, Ehab Mehmood, Rashid “A Novel Hybrid Textual-graphical Authentication Scheme With Better Security, Memorability, And Usability “
3. Takayuki Kawamura , Tadashi Ebihara , Naoto Wakatsuki , And Keiichi Zempo , “Eyedi: Graphical Authentication Scheme Of Estimating Your Encodable Distorted Images To Prevent Screenshot Attacks”

LITRATURE REVIEW

4.Syed Shabih Ul Hasan; anwar Ghani; muhammad Bilal; alireza Jolfae “Multifactor Pattern Implicit Authentication”

5.Ruben Tolosana; Ruben Vera-rodriquez; Julian Fierrez; Javier Ortega-garcia :
Touchscreen Password Biometrics Using Time-aligned Recurrent Neural Networks

6.Ba Ajeethra; Sv Gautham Prasath; R Arun Balaji; Kakelli Anil Kumara” Cryptography Based Face Authentication System For Secured Communication”

7.N.R. Pradeep; J Ravi “An Efficient Machine Learning Approach For Fingerprint Authentication Using Artificial Neural Networks”

8. Yang-waichow ,Wilysusilo, Jianfengwang, Richardbuckland, Joonsangbaek, Jongkilkim, Nanli”

<https://www.sciencedirect.com/science/article/pii/S1084804520303040>”

LITRATURE REVIEW

9. Chow, Yang-wai,susilo, Willy, Yang, Guomin,au, Man Ho, Wang, Cong
“Authentication And Transaction Verification Using Qr Codes With A Mobile Device”
10. Abdelouahid Derhab; Mohamed Belaoued; Mohamed Guerroumi;
Farrukh Aslam Khan”two-factor Mutual Authentication Offloading For Mobile Cloud Computing”
11. Yossi Oren; Dan Arad “Toward Usable And Accessible Two-factor Authentication Based On The Piezo-gyro Channel”
12. Petrova, K. Romaniello, Beatriz, Medlin, B.Vannoy, Sandra
“https://www.researchgate.net/publication/307879267_qr_codes_advantages_and_dangers”

MEATHOD DESCRIPTION

**SIMPLE
PASSWORD**

GRAPHICAL

TOKEN

BIOMETRICS

QR

**TWO
FACTOR**

MODULE WORK FLOW

EXPLANATION

Methods -1:

At first for our research, we collect around two to three research papers for each of the six authentication systems. Then using the research papers of each authentication systems, we found each of their limitations, strengths and usability and we collated each research papers key features and formed notes.

Methods -2:

Then after using the formed notes, we created a separate not by differentiating and evaluating each research papers according to some three factors namely usability, security and deploy-ability. Then we created a graph with x-axis as usability and y-axis as security and color code to differentiate the deploy-ability and then we plotted the six authentication systems and found out which has high

MODULE WORK FLOW

EXPLANATION

usability, security, deploy-ability and then we finally found the best authentication system as which has the highest points.

Methods -3:

We also have then implemented some of the authentication systems to analyze the real time scenario of the authentication system. For that we first learned the algorithm of the authentication system and then we searched on some webpages and YouTube videos and then using C++, Python, Android Studio and with some help of external modules and websites we created some implementations of authentication systems.

TECHNICAL IMPLEMENTATION AND CODING

SIMPLE PASSWORD

```
1 // Online C++ compiler to run C++ program online
2 #include <bits/stdc++.h>
3 using namespace std;
4
5 int main() {
6
7     cout<<"signup -01 or login-02";
8     int x;
9     cout<<"choose 1 or 2\n";
10    cin>>x;
11
12    map<string, string> mp;
13
14    if(x==1){
15        string x;
16        cout<<"enter a username to signup ";
17        cin>>x;
18
19        string y;
20        cout<<"enter a password to signup ";
21        cin>>y;
22
23        mp.insert({ x, y });
24        cout<<"signup succesfully done: now can now login using your
        credentials \n";
```

```

25         x=2;
26
27     }
28
29     string x1;
30     cout<<"enter a username for login  ";
31     cin>>x1;
32
33     string y1;
34     cout<<"enter a password for login  ";
35     cin>>y1;
36
37     bool flag = true;
38
39     if(flag == true){
40         for (auto itr = mp.begin(); itr != mp.end(); ++itr) {
41             if(itr->first == x1 ){
42                 if(itr->second == y1){
43                     cout<<"you are succesfully logged in";
44                 }else{
45                     cout<<"Username or password is incorrect: Login Failed"
46
47                 }
48             }else{
49                 cout<<"Username or password is incorrect: Login Failed";
50             }
51         }
52
53
54
55     }
56
57
58
59
60
61     return 0;
62 }

```

```
/tmp/ex10V1DG54.o
```

```
signup -01 or login-02choose 1 or 2
```

```
1
```

```
enter a username to signup rohan
```

```
enter a password to signup rohan132
```

```
signup succesfully done: now can now login using your credentials
```

```
enter a username for login rohan
```

```
enter a password for login rohan930
```

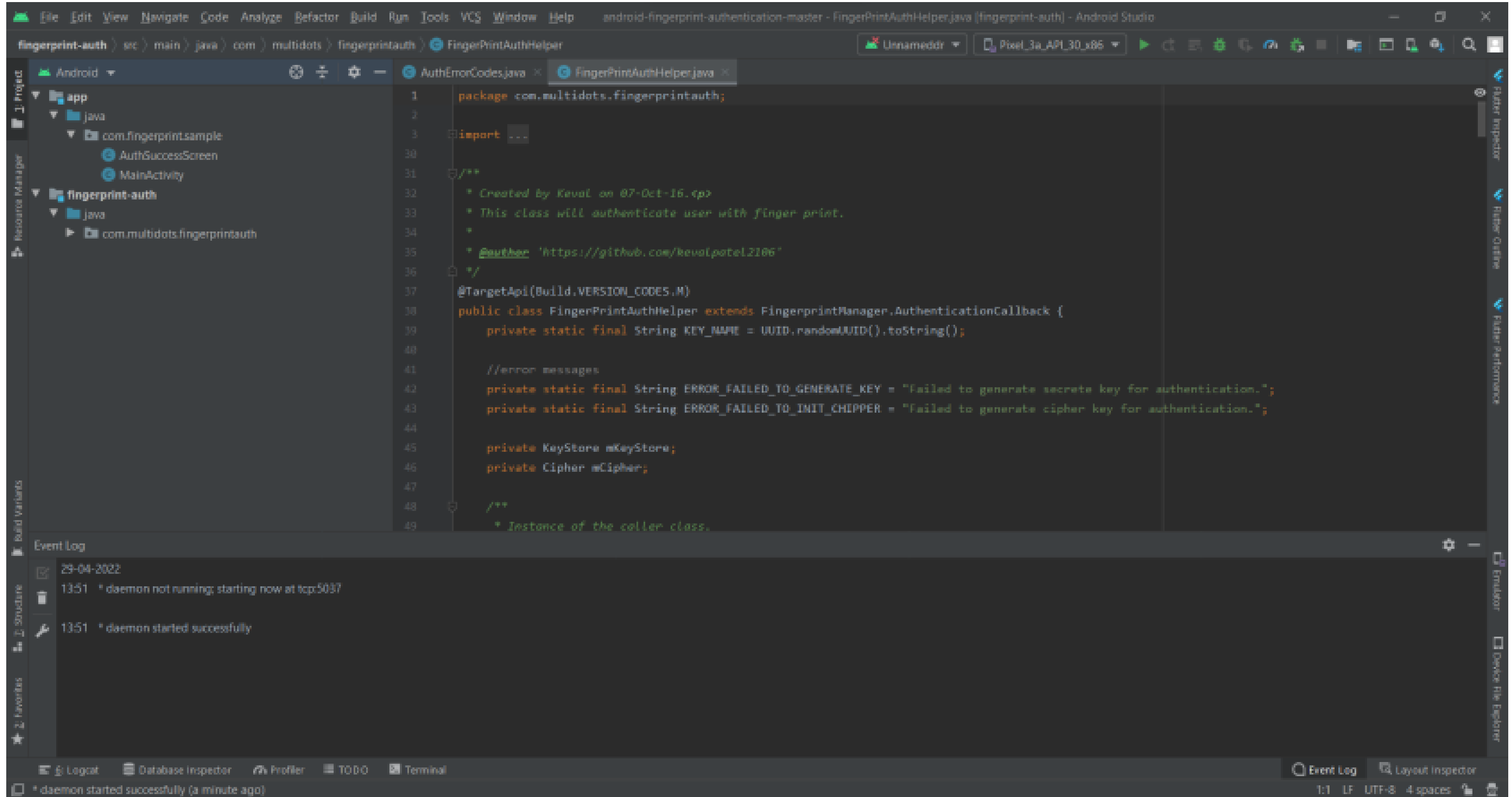
```
Username or password is incorrect: Login Failed
```

QR CODE BASED

```
qr.py > ...
1 import qrcode
2 import cv2
3
4
5 # creating a qr code that has the below given data
6 img= qrcode.make("this is the implementation of qr code")
7 # data will be saved in the new file named bit2.jpg
8 img.save("bit2.jpg")
9
10
11
12 # decoding the qr code which requires the qr code scanner
13
14 d= cv2.QRCodeDetector()
15 # d.detectAndDecode(cv2.imread("bit.jpg"))
16 val,points,straight_qrcode = d.detectAndDecode(cv2.imread("bit.jpg"))
17 print(val)
18
19
```



FINGERPRINT BASED



1:57

• VoLTE 7.0 K/s

FingerPrintAuthHelper



Scan your finger

1:59

• VoLTE 3.5 K/s







FingerPrintAuthHelper

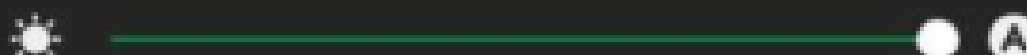
Authentication successful.


Authentication succeeded.

OTP BASED PASSWORD


```
1  # RECEIVING OTP using twillio
2
3  import random
4  from twilio.rest import Client
5
6  # // using my client twillio client id to send otps
7
8  otp = random.randint(1000,9999)
9  account_sid = "AC97057989246d93d0256e7872426c3e4a"
10 auth_token = 'bf2134bab0a1be59d86c7fc50c3fff7b'
11 client = Client(account_sid,auth_token)
12
13
14
15
16 msg = client.messages.create(
17
18     body = f"your otp is {otp}",
19     from_ = "+19206774932",    #this is yours own twillio phone number given after account creation
20
21     to = "+918770792399"    #to any phone number
22
23
24 )
25
```









Truecaller • SMS from 57575791 • 1m






3 7 6 0
OTP • 57575791


Copy

Ⓢ Don't share your OTP with anyone

MARK AS READ


DISMISS



Screen Mirror • 4m

Active

Screen mirroring is active



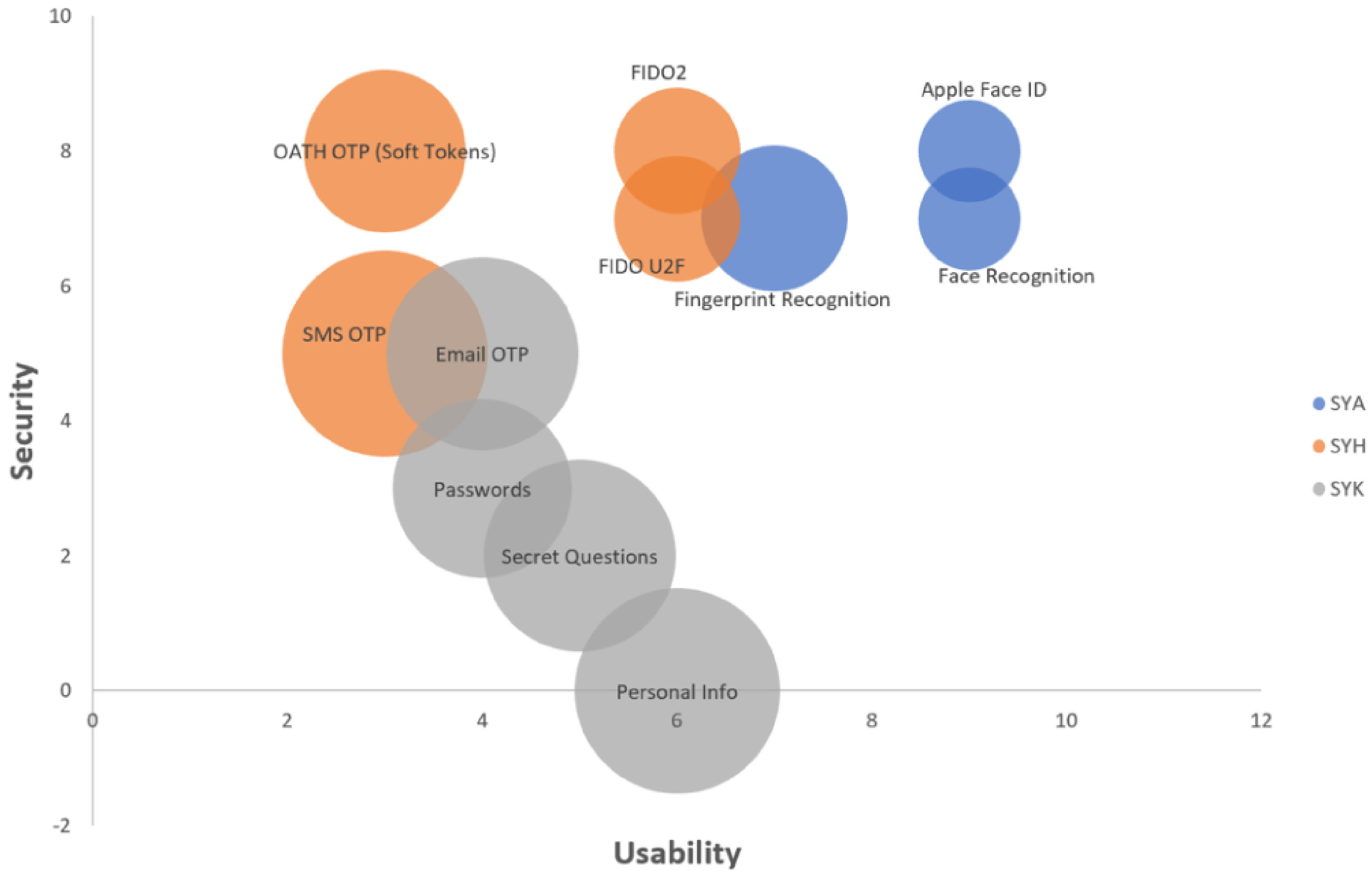
DEMO VIDEO

The working of all the implementations is shown in a demo video. The link to the demo video is given here

<https://youtu.be/7k7C49A96pE>

COMPARING AND FINDING THE BEST





CONCLUSION

In reference to the given diagram Apple face ID based authentication system is the most secure and convenient to use. Simple password is the least secure system. The term SYA – Something You Are; SYH – Something You Have; SYK – Something You Know are used in the chart. We can conclude from the research that the token based and the two factor authentication can be regarded as the best possible authentication methods today considering the security factor today.

THANK YOU