International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 2 Issue: 5 1126– 1129

Generation of Virtual Identities to Provide Secure Authentication through Mixed Fingerprint

Geetha.A[#1], Kanthasamy.D[#2], Anantharaj.B[#3]
[#1]*PG Student,* [#2]*Assistant Professor,* [#3]*Associate Professor*
Thiruvalluvar College of Engineering and Technology
Vandavasi.
E-Mail: gblueshine@gmail.com

Abstract— Mixing Fingerprints technique is one of the biometric techniques, which is used to provide more security for the particular user to access their own application. In the registration process, the user has to provide two different fingerprints and these fingerprints are mixed to form a new mixed fingerprint image. This image acts as an (original) virtual identity for the user Biometric authentications systems are gaining wide-spread popularity in recent years due to the advances in sensor technologies as well as improvements in the matching algorithms that make the systems to secure.. In here, we are using mixed fingerprint One Time Password technique to provide more security for access the application. The user has to provide two different fingerprints while registration process and these fingerprints are merged to form mixed fingerprint, and get OTP from Server. Then during the login process, the user has to provide both their fingerprints and it will be mixed again and compared with the original image. If the fingerprint and OTP is valid then the user is allowed to access the application.

Keywords— Fingerprints, Virtual Identities, decomposition, privacy protection, One Time Password.

*****

I.INTRODUCTION

Image processing is the process of converting image from one format to another format and gets some useful information from that. Image processing includes the following three steps.

1. Importing the image with optical scanner or by Digital photography
2. Analyzing and manipulating the image which includes data compression and image enhancement and spotting patterns that are not to human eyes like satellite photograph.
3. Output is the last stage in which result can be altered image or report that is based on image analysis.

Fingerprint in its narrow sense is an impression left by the friction ridges of a human finger. In a wider use of the term fingerprints are the traces of an impression from the friction ridges of any part of a human or other primate hand.

Fingerprint characteristics are named for their general visual appearance n and patterns. These are

Called Loops, Whorls and arches. Above 65 percent of the total population has loops, 30 percent have whorls, and 5 percent have arches.

A.FVC

The Fingerprint Verification Competition (FVC) is an international competition focused on fingerprint verification software assessment. A subset of fingerprint impressions acquired with various sensors was provided to registered participants, to allow them to adjust the parameters of their algorithms.

Arches have ridges that enter from one side of the fingerprint and leave from the other side with a rise in the center, whorls look like a bull's-eye, with two deltas (triangles). Loops enter from either the right or left and exit from the same side they enter.

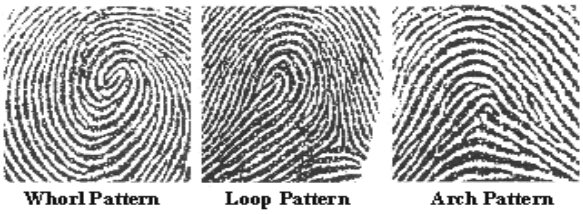

Figure.1 Fingerprint Characteristics

The rest of the paper is organized as follows; Section II presents the proposed approach for mixing fingerprint, Section III Algorithm used, IV Conclusion.

II .FINGERPRINT RECONSTRUCTION

[1] Fingerprint matching system use four types of representation schemes, [a]grayscale, [b]phase image, [c] Skelton image and [d] minutiae.

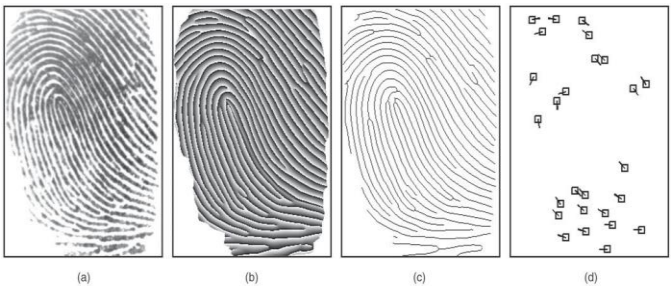

Figure.2 Fingerprint matching schemes.

The compactness of minutiae representation has created an impression that the minutiae template does not contain sufficient information to allow the reconstruction of the original grayscale fingerprint image.

The fingerprint reconstruction scheme is based on converting the minutiae representation to the phase representation. The phase is composed to continuous and the spiral phase. A reconstructed fingerprint is obtained by reconstructing the continuous phase and combining the continuous phase with the spiral phase.

[9] This Paper required enhancing the performance due to mixed fingerprints by exploring alternate algorithms for selecting and mixing the different pairs.

Reconstruction used for improving the interoperability among minutiae encoders and match from different vendors.

III MIXED FINGERPRINT

To [2] mix two fingerprints; each fingerprint pattern is decomposed into two different components, viz the continuous and spiral components. After pre aligning the components of each fingerprint the continuous components of one fingerprint is combined with spiral component of the other fingerprint. Mixing of fingerprint can be used to generate a large set of virtual identities.

The proposed approach, the ridge flow of a fingerprint can be represented as 2-D Amplitude and frequency (AM-FM).

$$I(x,y)=a(x,y)+b(x,y)\cos(¥(x.y)+n(x,y))$$

Where I(x,y) is the intensity of the original image at (x,y), a(x,y) is the intensity offset, b(x,y) is the amplitude, ¥(x,y) is the phase and n(x,y) is the noise.

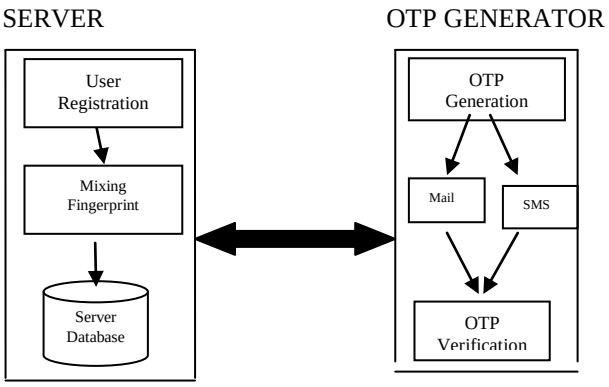

Figure 3. System Architecture

Mixing of two fingerprint improve the security level while accessing the application with the help of one time password the security level will be increased. Two algorithms can be used to achieve the security 1.Secure hash function 2. Secure random key generation algorithm.

IV SYMMETRIC HASH FUNCTION

The fingerprint verification process [3] includes the symmetric hash function value. Based on the symmetric hash values the fingerprints are verified. Online application process user have to register with their two fingerprint and the hash values are generated that should be stored to the hash database.

The fingerprint hash values are merged and stored to the database. During the login time the same fingerprint received from the client and merged with their fingerprint again the hash values are generated and compare with stored values from the database if the fingerprint value is valid the client will access the application.

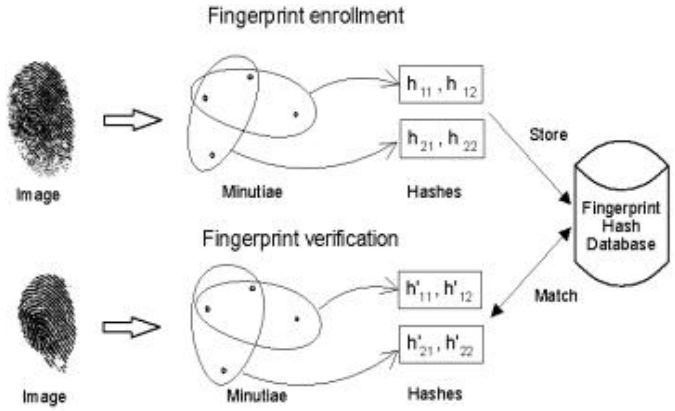

Figure 4. Application of hash function

The transformation of one fingerprint to the other can be described by the complex function f (z) =rz+t. specifically given n minutia point {c1, c2… cn} we construct following *m* symmetric hash function.

$$h_1(c_1, c_2, \ldots, c_n) = c_1 + c_2 + \cdots + c_n$$
$$h_2(c_1, c_2, \ldots, c_n) = c_1^2 + c_2^2 + \cdots + c_n^2 \quad (1)$$
.....
$$h_m(c_1, c_2, \ldots, c_n) = c_1^m + c_2^m + \cdots + c_n^m$$

Suppose that the another image of the fingerprint is obtained through above described transformation f(z) = rz +t, thus locations of corresponding minutia points are c0i = f(ci) = rci + t.

Hash functions of the transformed minutiae can be rewritten as

$$h_1(c_1, c_2, \ldots, c_n) = c_1 + c_2 + \cdots + c_n$$
$$= (rc_1 + t) + (rc_2 + t) + \cdots + (rc_n + t)$$
$$= r(c_1 + c_2 + \cdots + c_n) + nt = rh_1(c_1, c_2, \ldots, c_n) + nt$$
$$h_2(c_1, c_2, \ldots, c_n) = c_1^2 + c_2^2 + \cdots + c_n^2$$
$$= (rc_1 + t)^2 + (rc_2 + t)^2 + \cdots + (rc_n + t)^2$$
$$= r^2(c_1^2 + c_2^2 + \cdots + c_n^2) + 2rt(c_1 + c_2 + \cdots + c_n) + nt^2$$
$$= r^2h_2(c_1, c_2, \ldots, c_n) + 2rh_1(c_1, c_2, \ldots, c_n) + nt^2 \quad (2)$$

Let us denote the hash values of the minutia set of one fingerprint as hi =hi(c1, c2, . . . , cn) and hash values of corresponding minutia set of another fingerprint as h0i = hi(c01, c02, . . . , c0n).

Equations 2 now become

$$h_1 = rh_1 + nt$$
$$h_2 = r^2h_2 + 2rth_1 + nt^2$$
$$h_3 = r^3h_3 + 3r^2th_2 + 3rt^2h_1 + nt^3 \quad (3)$$
$$\ldots$$

Equations 3 have two unknown variables r and t. If we take into account errors introduced during fingerprint scanning and minutia search, the relation between hash values of enrolled fingerprint $\{h_1, \ldots, h_m\}$ and hash values of test fingerprint $\{h_1, \ldots, h_m\}$ can be represented as

$$h_i = f_i(r, t, h_1, \ldots, h_n) + \quad i \quad (4)$$

The matching between hash values of enrolled fingerprint $\{h_1, \ldots, h_m\}$ and hash values of test fingerprint $\{h_1, \ldots, h_m\}$ consists in finding r and t that minimize errors $_i$. During algorithm implementation we considered minimization of error functions = $\alpha_i|_i|$, where weights α_i were chosen empirically.

A.ONE TIME PASSWORD GENERATOR

This OTP is based on the very popular algorithm HMAC SHA. The HMAC SHA is an algorithm generally used to perform authentication by challenge response. It is not an encryption algorithm but a hashing algorithm that transforms a set of bytes to another set of bytes. This algorithm is not reversible which means that you cannot use the result to go back to the source.

A HMAC SHA uses a key to transform an input array of bytes. The key is the secret that must never be accessible to a hacker and the input is the challenge. This means that OTP is a challenge response authentication.

The secret key must be 20 bytes at least; the challenge is usually a counter of 8 bytes which leaves quite some time before the value is exhausted.

The algorithm takes the 20 bytes key and the 8 bytes counter to create a 8 digits number. This means that there will obviously be duplicates during the life time of the OTP generator but this doesn't matter as no duplicate can occur consecutively and an OTP is only valid for a couple of minutes.

B.OTP Strong authentication method

- The key is 20 digits
- A password is a couple counter/password, only valid once and a very short time
- The algorithm that generates each password is not reversible
- With an OTP token, the key is hardware protected
- If the OTP is received on your phone, the key always stays at the server

Those few characteristics make the OTP a strong authentication protocol. The weakness in an authentication is usually the human factor. It is difficult to remember many complex passwords, so users often use the same one all across the internet and not really a strong one.

With an OTP, you don't have to remember a password, the most you would have to remember would be PIN code (4 to 8 digits) if the OTP token is PIN protected. In the case of an OTP sent by a mobile phone, it is protected by your phone security. A PIN is short but you can't generally try it more than 3 times before the token is locked.

An OTP depends on 2 parameters:

- A secret key
- A counter

Even if a hacker intercepts millions of OTP the algorithm is not reversible which means that even if you know the key you can't go back to the counter that was used to generate the OTP. So without the key and the counter, it is virtually impossible even with millions of OTP to find a pattern to guess the key and the current counter value.

OTP are usually used to perform authentication or to verify a transaction with a credit card. In the case of a transaction an OTP is sent to the mobile phone of the user, for an authentication if is possible to use either a secure token or to request an OTP to be send to the user phone.

V .RESULTS

Need for implementing the paper is to provide the security for accessing on online shopping applications using mixed fingerprint(Figure 5) and OTP..Here in this we implemented modules of user registration, data stored server database and verify the fingerprint and OTP.In this module We will create an application to User registration. For creating an Application the design fields like Name, DOB, Email, Phone number, Fingerprint,. Once the application created the user is allowed to access the application.Server is used to store the data to database and verify the fingerprint during the login time and allow accessing the application

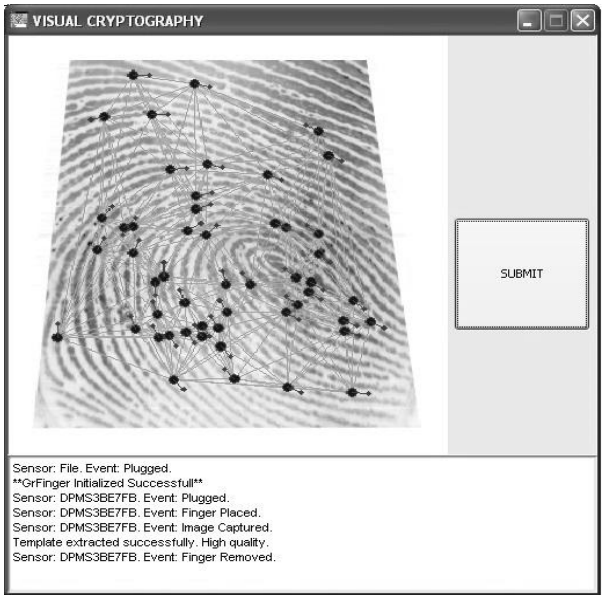

Figure 5 FINGERPRINT GENERATIONS

This is usually the authentication method used when a transaction is verified with an OTP. The bank system sends you an OTP and you then have few minutes to enter this OTP. This mechanism doesn't need any synchronization process as the OTP is originally generated by the server and send to a third party device. The server expects that you type the correct OTP within generally 2 mns. If you fail to do it, you just ask a new OTP and then enter it within the given time.

When a system supports both authentication methods, it means that the back-end has 2 different keys and counters; one pair for the OTP token and one pair for the OTP transmitted by SMS.

A .Using an OTP token

The original product I worked on when we implemented one of the first versions of the OTP in a Java card was using an OTP token with a screen or a mobile phone with a card applet to generate the OTP. In this model both the server and the authentication token have to generate an OTP that must be synchronized. The process is the following: The user generates an OTP with his token, type it and press OK. The server receives the OTP generated by the token, it increments the counter and generates a new OTP.

This is where there is a possible synchronization issue.

Synchronization issues

If the user enters the correct OTP, then the server when it increments the counter and calculate the OTP, the authentication will be successful.

Desynchronization could arise:

1. The user accidentally press the generate button of his token and doesn't perform an authentication. In this case the counter of the token would be ahead of the server counter by few steps.
2. The user enters an OTP without generating it from the token. In this case the counter of the server would be ahead of the token counter.
3. The user generates an OTP with the token but types a wrong OTP.

VI CONCLUSION

In this word demonstrate the concept of mixing fingerprint and one time password for accessing the on line application. Fingerprints are decomposed into two components spiral and continuous components. These components are merged and stored to the database. Using fingerprint technology the security level for accessing the application increase the security level using OTP concept.

REFERENCES

[1] Asem Othman and Arun Rose, " On Mixing Fingerprint", IEEE Transaction on nformation referensics and security, vol.8, No.1, Jan,13.

[2] J.Feng and A.K.Jain, "Fingerprint reconstruction: From minutiae to phase, "IEEE Trans, pattern Anal. Mach. Intell., vol.33, no.2, pp, 209-223, Feb. 2011.

[3] Asem Otheman and Arun Ross, " Mixing Fingerprints For Generating Virtual on Information Forensics and Security (WIFS), (Forz do lguack Bracil)Nov/Dec 2011.

[4] Sergey Tulyakov, Faisal Farooq and venu Govindaraju, "Symmetric Hash Function Minutiae", SUNY at Buffalo NY 14228, USA.

[5] Megha Kulsherstha, Pooja, V.K Banga," Selection of an Optimal Algorithm for Fingerprint Matching", World Academy of Science, Enineering and Technology 51 2011.

[6] Jianjiang Feng and Anil K.Jain, " FM Model Based Fingerprint Reconstruction from Minutiae Template", Department of Computer Science and Engineering Michigan State University.

[7] D.Malton, D.Maio, A.Jain and S.Prabhakar(2009). Handbook of Finerprint Recognition, New York: Springer, Verlag, New York Inc.

[8] R. Chppelli, A.Lumini, D.Maio and D.Maltoni(2007),"Fingerprint image reconstruction from standard templates," IEEE Trans . Pattern Anal, Mach Intell vol 29, No.9, pp.1489-1503.

[9] A.Ross, J.Shah and A.Jain(2007), "From Template to image: Reconstructing fimgerprints from minutiae points ," IEEE Trans, Pattern Anal, Mach, Intell., Vol.29, No.4, pp 544-560.

Ms A.Geetha received her M.SC. Degree in Information Technology (IT) in 2008 from Bharathidasan University and post graduate degree in Computer Science and Engineering from Anna University Chennai, India. Her areas of interest are Java, operating systems and Computer Networks. She has presented many papers in national conferences in various fields. As part of this paper, she is working on Secure Authentication through Mixed Fingerprint. She is a member of ISTE.