International Journal of Computer Sciences and Engineering Open Access
Survey Paper Volume-2, Issue-5 E-ISSN: 2347-2693

A Survey on Secure Crypto-Biometric System using Blind Authentication Technique

Anil S Naik [1*], Shivappa M Metagar [2] and Praveenkumar D Hasalkar [3]
[1*,2,3]*AsstProf, (IT), WIT, Solapur University, India*

www.ijcaonline.org

Received: 18 Apr 2014 Revised: 10 May 2014 Accepted: 22 May 2014 Published: 31 May 2014

***Abstract*—** Reliable user authentication is becoming an increasingly important task in the Web-enabled world. Biometrics-based authentication systems offer obvious usability advantages over traditional password and token-based authentication schemes. However, biometrics also raises some issues in lack of privacy, template security, and revocability. The use of cryptographic primitives to bolster the biometric authentication system can solve the issues in biometric system.The combination of biometrics over cryptography may lead to a problem of lack of accuracy in biometric verification. In this paper, We propose a cryptographic protocol for biometrics authentication without revealing personal biometrical data against malicious verifier the protocol is termed as blind biometric authentication protocol, which addresses the concerns of user's privacy, template protection, trust issue. The accuracy problem can be solved by designing a classifier. The protocol is blind in the sense that it reveals only the identity, and no additional information about the user or the biometric to the authenticating server or vice-versa. The proposed protocol is secure to different attacks.

***Keywords*—** Biometrics, Cryptosystems, Privacy, Public Key Cryptography, Security, Authentication

I. INTRODUCTION

Today's human authentication factors have been placed in three categories, namely What you know, e.g password, secret, personal identification number (PIN); What you have, such as token, smart card etc. and What you are, biometrics for example. However, the first two factors can be easily fooled. For instance, password and PINs can be shared among users of a system or resource. Moreover, password and PINs can be illicitly acquired by direct observation. The main advantage of biometrics is that it bases recognition on an intrinsic aspect of a human being and the usage of biometrics requires the person to be authenticated to be physically present at the point of the authentication. These characteristics overcome the problems whereas password and token are unable to differentiate between the legitimate user and an attacker.

In addition biometric authentication information cannot be transferred or shared; it is a powerful weapon against repudiation. However, it also suffers from some inherent biometrics-specific threats [1]. A hacker who gains physical or remote access to an authentication server can steal the stored templates, which are non replaceable in case of plain templates. Concerns are also on the privacy as many biometrics reveal personal information beyond just identity. Widespread use of biometric authentication also provides the ability to track a person through every activity in his life, which introduces another significant privacy concern. The primary concerns in widespread use of biometrics for remote and onsite authentication are in i) template protection, ii) privacy of the user, iii) trust between user and server, and iv) network security. The ideal solution to overcoming all the privacy and security concerns would be to apply a strong encryption on the biometric samples as well as the classifier parameters, and carry out all the computations in the encrypted domain.

However, the primary goal of a strong encryption algorithm is to destroy any pattern that would be present in the data. We now need to carry out a pattern classification task (identity verification) in the encrypted domain. These two goals are contradictory. In other words, security/privacy and accuracy seems to be opposing objectives. Different secure authentication solutions achieve their goal through a compromise between privacy and accuracy or by making restrictive assumptions on the biometric data. The primary difference in our approach is that we are able to design the classifier in the plain feature space, which allows us to maintain the performance of the biometric itself, while carrying out the authentication on data with strong encryption, which provides high security/privacy. However, such a solution would require an *algebraic homomorphic encryption* scheme [2]. The only known doubly homomorphic scheme has recently been proposed by Gentry [3] and would mostly lead to a computationally intensive theoretical solution. We show that it is possible to achieve a practical solution using distribution of work between the client (sensor) and the server (authenticator), using our proposed randomization scheme.

II. BLIND AUTHENTICATION

We define *Blind Authentication* as "a biometric authentication protocol that does not reveal any information about the biometric samples to the authenticating server. It also does not reveal any information regarding the classifier, employed by the server, to the user or client." Blind authentication, proposed in our paper, is able to achieve both strong encryption-based security as well as accuracy of a powerful classifier. While the proposed approach has

similarities to the blind vision [4] scheme for image retrieval, it is far more efficient for the verification task.

Blind Authentication addresses all the concerns mentioned Before 1) The ability to use strong encryption addresses template protection issues as well as privacy concerns.

2) Non-repudiable authentication can be carried out even between nontrusting client and server using a trusted third party solution.

3) It provides provable protection against replay and clientside attacks even if the keys of the user are compromised.

4) As the enrolled templates are encrypted using a key, one can replace any compromised template, providing revocability, while allaying concerns of being tracked. In addition, the framework is generic in the sense that it can classify any feature vector, making it applicable to multiple biometrics. Moreover, as the authentication process requires someone to send an encrypted version of the biometric, the nonrepudiable nature of the authentication is fully preserved, assuming that spoof attacks are prevented.

We assume that authentication is done through a generic *linear classifier*. One could use any biometric in this framework as long as each test sample is represented using a feature vector of length n . Note that even for biometrics such as fingerprints, one can define fixed length feature representations [5]. Let ω be the parameters of the linear classifier (perceptron). The server accepts the claimed identity of a user, if , $\omega.x < \tau$ where τ is a threshold.As we do not want to reveal the template feature vector (ω) or the test sample (x) to the server, we need to carry out the perceptron function computation directly in the encrypted domain. Computing ω.x involves both multiplication and addition operations, thus computing it in the encrypted domain requires the usage of a doubly homomorphic encryption scheme [6]. In the absence of a practical doubly homomorphic encryption scheme (both additive and multiplicative homomorphic), our protocol uses a class of encryption that are multiplicative homomorphic, and we simulate addition using a clever randomization scheme over one-round of interaction between the server and the client. An encryption scheme E(x) is said to be multiplicative homomorphic, if E(x).E(y)=E(xy) for any two numbers x and y . We use the popular MD5 encryption scheme , which satisfies this property.

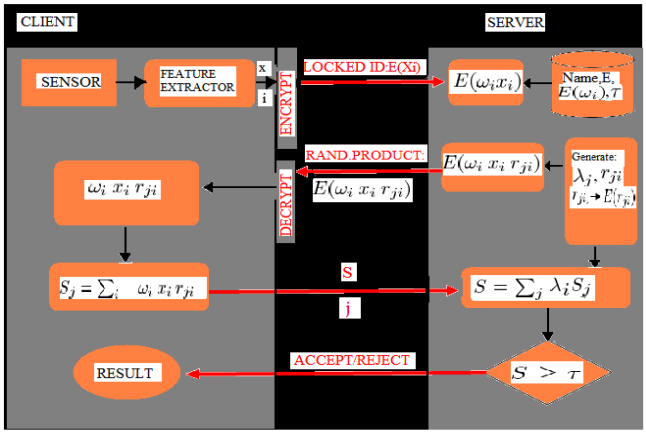

Fig 1. Blind authentication process.

An overview of the authentication process is presented in Fig. 1. We assume that the server has the parameter vector ω in the encrypted form, i.e E(ω) , which it receives during the enrollment phase. The authentication happens over two rounds of communication between the client and the server. To perform authentication, the client locks the biometric test sample using her public key and sends the *locked ID* to the server. The server computes the products of the locked ID with the locked classifier parameters and randomizes the results. These *randomized products* are sent back to the client. During the second round, the client unlocks the randomized results and computes the sum of the products. The resulting *randomized sum* is sent to the server. The server derandomizes the sum to obtain the final result, which is compared with a threshold for authentication. As we described before, both the user (or client) and the server do not trust each other with the biometric and the claimed identity. While the enrollment is done by a trusted third party, the authentications can be done between the client and the server directly. The client has a biometric sensor and some amount of computing power. The client also possesses an MD5 private– public key pair, and . We will now describe the authentication and enrollment protocols in detail.

A. Authentication

We note that the computation of requires a set of scalar multiplications, followed by a set of additions. As the encryption used is homomorphic to multiplication, we can compute, , at the server side[9].

However, we cannot add the results to compute the authentication function. Unfortunately, sending the products to the client for addition will reveal the classifier parameters to the user, which is not desirable. We use a clever randomization mechanism that achieves this computation without revealing any information to the user[7]. The randomization makes sure that the client can do the summation, while not being able to decipher any information from the products. The randomization is done in such a way that the server can compute the final sum to be compared with the threshold. The overall algorithm of the authentication process is given in Algorithm 1. Note that all the arithmetic operations that we mention in the encrypted domain will be -operations, i.e., all the computations such as *(a op b)*

will be done as *(a op b) mod q* , where q is defined by the encryption scheme employed.

Algorithm 1: Authentication

1: Client computes feature vector, $x_{1..n}$, from test data
2: Each feature x_i is encrypted $(E(x_i))$ and sent to server
3: Server computes $kn + k$ random numbers, r_{ji} and λ_j, such that, $\forall_i$, $\sum_{j=1}^{k} \lambda_j \, r_{ji} = 1$
4: Server computes $E(\omega_i \, x_i \, r_{ji}) = E(\omega_i) \, E(x_i) \, E(r_{ji})$
5: The kn products thus generated are sent to the client
6: The client decrypts the products to obtain: $\omega_i \, x_i \, r_{ji}$
7: Client returns $S_j = \sum_{i=1}^{n} \omega_i \, x_i \, r_{ji}$ to the server
8: Server computes $S = \sum_{j=1}^{k} \lambda_j \, S_j$
9: **if** $S > \tau$ **then**
10: return *Accepted* to the client
11: **else**
12: return *Rejected* to the client
13: **end if**

In this Algorithm the server carries out all its computation in the encrypted domain and hence does not get any information about the biometric data(x). The server has an access to a random number generator.

One can deal with variable length features and warping – based matching techniques using a similar approach. The authentication process thus maintains a clear separation information between the client and server, and provides complete security to user.

B. Enrollment

Algorithm 2: Enrollment

1: Client collects multiple sample of her biometric, $B_{1...k}$
2: Feature vectors, x_i, are computed from each sample
3: Client sends x_i, along with her identity and public key, E, to the enrollment server
4: Enrollment server uses x_i and the information from other users to compute an authenticating classifier (ω, τ) for the user
5: The classifier parameters are encrypted using the users public key: $E(\omega_i)$
6: $E(\omega_i)s$, along with the user's identity, the encryption key (E), and the threshold (τ), are sent to the authentication server for registration
7: The client is then notified about success

During enrollment the client send samples of her biometric to the enrollment server.

The trained parameters are encrypted and sent to the authentication server and a notification is sent back to the client.

An ideal biometric system would ensure privacy and hence need not demand any trust, thus making it possible for large set of applications.

III. SECURETY ISSUES

Security of the system refers to the ability of the system to withstand attacks from outside to gain illegal access or deny access to legitimate users. Since we are dealing with insecure networks, we are primarily concerned with the former[8]. In terms of information revealed, security is related to the amount of information that is revealed to an attacker that would enable him to gain illegal access. Privacy on the other hand is related to the amount of user information that is revealed to the server. Ideally, one would like to reveal only the identity and no additional information. Most of the current systems provide very little privacy, and hence demands trust between the user and the server. An ideal biometric system would ensure privacy and hence need not demand any trust, thus making it applicable in a large set of applications .We now take a closer look at the security and privacy aspects of the proposed system.

A. SYSTEM SECURITY

Biometric systems are known to be more secure as compared to passwords or tokens, as they are difficult to reproduce. As the authentication process in the proposed system is directly based on biometrics we gain all the advantages of a generic biometric system. The security is further enhanced by the fact that an attacker needs to get access to both the user's biometric as well as her private key to be able to pose as an enrolled user [10].

1) Server Security: We analyze the security at the server end using two possible attacks on the server.
2) Client Security: At the client side, we will consider the fallowing attack scenarios .
3) Network Security :An insecure network is susceptible to snooping attacks .Let us consider

the following attack scenarios .

B. PRIVACY

Privacy, as noted before, deals with the amount of user information that is revealed to the server during the process of enrollment and authentication. We noted that there are two aspects of privacy to be dealt with:

1. Concern of revealing personal information*:* As the template or test biometric sample is never revealed to the server, the user need not worry that the use of biometrics might divulge any personal information other than her identity.

2. Concern of being tracked*:* One can use different keys for different applications (servers) and hence avoid being tracked across uses. In fact, even the choice biometric or real identity of the user itself is known only to the enrolling server.

The authenticating server knows only the user ID communicated by the enrollment server and the biometric is obtained in the form of an encrypted feature vector. As the user and server need not trust each other, the framework is applicable to a variety of remote and on-site identity verification tasks. Moreover, we note that there is no delegation of trust by the server to a program or hardware at the user's end, thus making it applicable to a variety of usage scenarios.

IV. IMPLEMENTATION AND ANALYSIS

Representation of negative numbers: Use an Implicit sign representation. Use (0, M/2) as positive and rest as negative. Sign conversion is carried out using additive inversion of Z. Overflow and Underflow: Operations are valid and correct as long as range of data is (-M/2, M/2). Integer Division and thresholding: RNS domain is finite and hence not all divisions are defined. Dividing integer A by B is defined as $A/B = (a_i.b_i^{-1})$ mod m_i Defining Equivalent operations: For every f(x), we need to define f`(x) such that merging f`(x_i) would give f(x).

Experiments designed to evaluate the efficiency and accuracy of proposed approach. For evaluation, an SVM based verifier based on client-server architecture was implemented. Accuracy: as no assumptions are made, accuracy remains same.

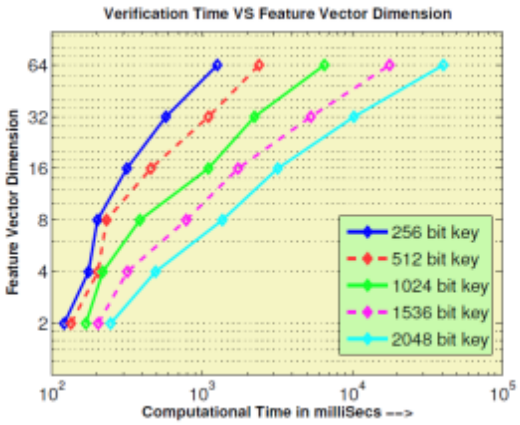

Fig 2. Verification time for various key sizes and feature vector lengths.

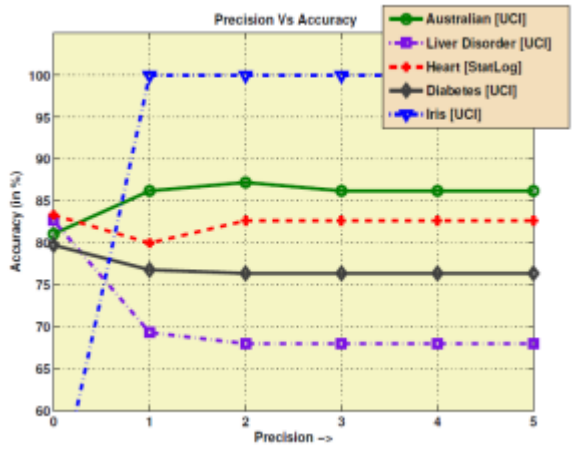

Fig 3. Variation of accuracy with respect to the precision of representation

ANALYSIS

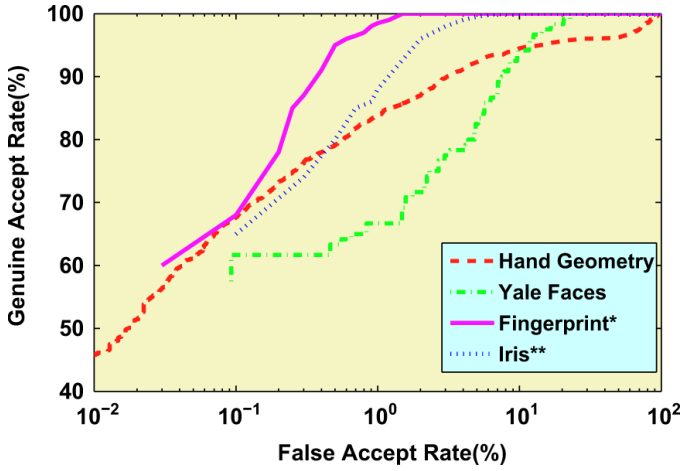

Fig.11. ROC curves for verification.

ROC curve implies the analysis of all the biometric methods used for providing the security to the important data. Here we can say that we doing the comparison in the various methods those are hand-geometry, face recognition, fingerprint and iris recognition.
The graph explains that as compared to all other methods fingerprint is the most recommendable. As we can see that figure print is most stable in graph.

V. CONCLUSION

The primary advantage of the proposed approach is that we are able to achieve classification of a strongly encrypted feature vector using generic classifiers. In fact, the authentication server need not know the specific biometric trait that is used by a particular user, which can even vary across users. Once a trusted enrollment server encrypts the classifier parameters for a specific biometric of a person, the authentication server is verifying the identity of a user with respect to that encryption. The real identity of the person is hence not revealed to the server, making the protocol, completely blind. This allows one to revoke enrolled templates by changing the encryption key, as well as use multiple keys across different servers to avoid being tracked, thus leading to better privacy. The proposed blind authentication is extremely secure under a variety of attacks and can be used with a wide variety of biometric traits. Protocols are designed to keep the interaction between the user and the server to a minimum with no resort to computationally expensive protocols such as secure multiparty computation (SMC) . As the verification can be done in real-time with the help of available hardware, the approach is practical in many applications. The use of smart cards to hold encryption keys enables applications such as biometric ATMs and access of services from public terminals.

REFERENCES

[1] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," IEEE Trans. Circuits Syst. Video Technol., vol. **14**, no. **1**, pp. **4–20**, Jan. **2004**.
[2] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," IBM Syst. J., vol. **40**, no. **3**, pp. **614–634**, Mar. **2001.**
[3] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Commun. ACM, vol. **21**, no. **2**, pp. **120–126, 1978**.
[4] C. Fontaine and F. Galand, "A survey of homomorphic encryption for nonspecialists," EURASIP, vol. **1**, pp. **1–15, 2007.**
[5] C. Gentry, "Fully homomorphic encryption using ideal lattices," STOC, pp. **169–178, 2009**.
[6] F. Farooq, R. M. Bolle, T.-Y. Jea, and N. Ratha, "Anonymous and revocable fingerprint recognition," in CVPR Biometrics Worshop, Jun. **2007**, pp. **1–7.**
[7] A. Teoh, D. Ngo, and A. Goh, "Biohashing: Two factor authentication featuring fingerprint data and tokenised random number," Pattern Recognit., vol. **37**, no. **11**, pp. **2245–2255**, Nov**. 2004**.
[8] A. K. Jain, K. Nandakumar, and A. Nagar, "Biometric template security," EURASIP, vol. **8,** no. **2**, pp. **1–17, 2008**.
[9] M. Upmanyu, A. M. Namboodiri, K. Srinathan, and C. V. Jawahar, "Efficient biometric verification in the encrypted domain," in 3rd Int. Conf. Biometrics, Jun. **2009,** pp. **906–915.**
[10] Yogesh Badhe, Hafij Balbatti, Neelkanth Kaladagi , Kranti Kumar, "IRIS Recognition and Authentication System for Enhancing Data Security" , International Journal of Computer Sciences and Engineering, Volume-**02**, Issue-**03**, Page No (**1-5**), March **2014**.

AUTHORS PROFILE

ANIL S NAIK received B.E.degree (Electronics and Communication Engineering) in 2009 from BEC, Bagalkot and M.Tech (Information Technology) in 2011 from AMCEC, Bangalore. He is presently Working as Assistant Professor in the department of IT, W.I.T Solapur, Maharastra. His research nterests are in the area of Software Engineering ,Networks, Network Security, Data Mining, Web Technology and Image Processing.

SHIVAPPA M METAGAR received B.E. degree (Computer Science & Engineering) in 2010 from KBNCE, Gulbarga and M.Tech (Digital Communication and Networking) in 2012 from BTLIT, Bangalore. He is presently Working as Assistant Professor in the department of CSE, W.I.T Solapur, Maharastra. His research interests are in the area of Networks, Network Security, Data Mining, Web Technology and Image Processing.

PRAVEENKUMAR D HASALKAR received B.E. degree (Computer Science & Engineering) in 2007 from SLN College of Engineering, Raichur and M.Tech (Computer Science & Engineering) in 2012 from BVB Hubli

He is presently Working as Assistant Professor in the department of CSE, W.I.T Solapur, Maharastra. His research interests are in the area of Networks, Network Security, Data Mining, Web Technology and Image Processing.