Biometric Authentication System using Non-Linear Chaos

Mr.A.Senthil Arumugam [#1], *Dr.N.Krishnan* [*2]

[1]Centre for Information Technology and Engineering, Manonmaniam Sundaranar University, Tirunelveli, India

vethathirisen@yahoo.co.in

[2] Centre for Information Technology and Engineering, Manonmaniam Sundaranar University, Tirunelveli, India

krishnan17563@gmail.com

Abstract— A major concern nowadays for any Biometric Credential Management System is its potential vulnerability to protect its information sources; i.e. protecting a genuine user's template from both internal and external threats. These days' biometric authentication systems face various risks. One of the most serious threats is the vulnerability of the template's database. An attacker with access to a reference template could try to impersonate a legitimate user by reconstructing the biometric sample and by creating a physical spoof. Susceptibility of the database can have a disastrous impact on the whole authentication system. The potential disclosure of digitally stored biometric data raises serious concerns about privacy and data protection. Therefore, we propose a method which would integrate conventional cryptography techniques with biometrics. In this work, we present a biometric crypto system which encrypts the biometric template and the encryption is done by generating pseudo random numbers, based on non-linear dynamics.

Keywords: Biometric Encryption, Tent Map, M-Logistic function

I. INTRODUCTION

Biometric methods are used in many Domains and for many purposes. Biometric authentication serves an individual to prove his or her authenticity. Biometric characteristics are uniquely associated with each user and thus represent the strongest form of personally identifiable information. Obviously this strengthens the authentication process; on the other hand the possibility that a biometric template could be stolen or exchanged raises concerns on its possible uses and abuses. It may be likely to get information about the enrolled person from their biometric template. It's also achievable to compromise any traditional biometric systems in order to gain access without presenting a biometric sample. In the same way, the efficacy of access control mechanisms is inherently limited, e.g. against internal attacks or in the presence of software vulnerabilities. In conventional cryptography, user authentication is based on possession of secret keys (such as a token or possession of smart card or remembering a password); such keys can be forgotten, lost, stolen, or may be illegally shared. So the biometrics and the conventional cryptography have their own potential vulnerabilities, but the ability to combine a cryptography and biometrics can enhance the trustworthiness of an authentication system.

(1) Threat Vectors: Issues & Challenges – Threat Vector is a path or a tool that an imposter uses to attack the biometric system. An attack is conducted by a threat agent, which is defined as person who, intentionally or otherwise, seeks to compromise the biometric system. Imposter: any person who intentionally or otherwise, poses as an authorized user. The imposter may be an authorized or unauthorized user. Attacker: Any Person or system attempting to compromise the biometric device. Motivation may include unauthorized entry or denial of service. Authorized user: any person or system admin to use the biometric system but who may unintentionally compromise the biometric device: meant for unintentional and human error, such as an administrator error in configuring a biometric system [2].

(2) False Enrollment using Fake Traits: The accuracy of the biometric data if founded on legitimate enrollments. If identity is faked, the enrollment data will be an accurate biometric of the individual but identity will be incorrectly matched. Spoofing or providing a fake physical biometric designed to circumvent the biometric system. This can be relatively easily conducted as little or no technical system knowledge is mandatory. The original biometric can be relatively easily obtained from many sources, with or without the permission and co-operation of the "Genuine User" of that biometric sample.

(3) Reuse of Residuals: Some biometric devices and systems may retain the last few biometrics extracted and templates used in local memory. If an attacker gain access to this data, they may be able to reuse it to provide a valid biometric. Clearing memory and eliminating identical sample being used consecutively is an effective security mechanism [2].

(4) Replay Attacks: In replay attacks, the data related to the presentation of a biometric is captured and replayed. Alternatively a false data stream is injected between the sensor and the processing system. A data stream representing a fake biometric is injected into the system. In most cases this will involve some physical tampering with the device. Where templates are stored on an RFID or proximity card, the data is likely to be unencrypted. This can assist the unauthorized collection of the data for later replay [2].

II. BIOMETRIC AUTHENTICATION AND BIOMETRIC RANDOM KEY GENERATION

Biometric Cryptosystem is the only solution to defeat all kind of threat vectors. Biometric crypto system combines cryptography and biometrics; while cryptography ensures high security and biometrics eliminates the need of carrying the tokens or remembering passwords. Biometric encryption is designed to avoid these problems by embedding the secret code into the template, in a way that can be decrypted only with a biometric sample of the enrolled individual. Since the secret code is bound to the biometric template, an attacker will not be able to determine either the enrolled biometric sample or secret code, even if they have access to the biometric software and hardware.

2.1. Biometric Application Programming Interfaces

The Biometric Application Programming Interface is intended to provide a high-level generic biometric authentication model; one suited for any form of biometric technology. It covers the basic functions of Enrollment, Verification, and Identification, and includes a database interface to allow a biometric service provider to manage the Identification population for optimum performance. It also provides primitives that allow the application to manage the capture of samples on a client, and the Enrollment, Verification, and Identification on a server. This specification defines the Application Programming Interface and Service Provider Interface for a standard biometric technology interface.

Application Level API is the high level at which the basic biometric functions are implemented - those which an application would generally use to incorporate biometric capabilities for the purpose of human identification. This standard uses the term template to refer to the biometric enrollment data for a user. The template must be matched within a specified tolerance by sample taken from the user, in order for the user to be authenticated. The term biometric identification record refers to any biometric data that is returned to the application; including raw data, intermediate data, and processed samples ready for verification or identification, as well as enrollment data. Typically, the only data stored persistently by the application is the biometric identification record generated for enrollment i.e., the template [3].

2.2. Enrollment & Verification using BioAPIs and PHP-AJAX

The purpose of enrollment is to construct a database of genuine users. It has to be somehow determined what makes a subject eligible to be enrolled, and all enrollees must be checked against these criteria. Biometric samples and other credentials are stored in the database, which in case of verification system might be a distributed / centralized database. Each subject is enrolled with a biometric template. The subject is issued some possession that contains the biometric template. There are three principal high-level abstraction functions in the API: (1) Enroll: Samples are captured from a device, processed into a usable form from which a template is constructed, and returned to the application. (2) Verify: One or more samples are captured, processed into a usable form, and then matched against an input template. The results of the comparison are returned. (3) Identify: One or more samples are captured, processed into a usable form, and matched against a set of templates [3]. Biometric Application Programming Interface supports PKI functionality through the Captured Biometric Application Programming Interface extension. This is particularly important when considering the use of PKI in the trusted device model. This model allows trusted devices to accept digital certificates from outside sources and encrypt and sign the data with their own certificates, making biometric devices perfect tools for authentication.

2.3. Biometric Cryptosystem

Biometric Cryptosystem is a new and exciting area combining the features from the fields of Biometrics and Cryptography. In biometric systems the integrity of data transmission must be secure all the way from the sensor to the application. This is typically achieved by cryptographic methods. In conventional cryptography, encryption is a mathematical process that helps to disguise the information contained in messages that is either transmitted or stored in a database, and there are three main factors that determine the security of any cryptosystem; the complexity of the mathematical process or algorithm, the length of the encryption key used to disguise the message, and safe storage of the key, known as key management [4, 5].

The enhancement of security level in biometrics-based systems can be done in two ways; use of encryption keys to protect biometric information or use of biometric mechanisms to secure the privacy of encryption keys and access to data. A biometric system always produces a Yes/No response, which is essentially one bit of information. Therefore, an obvious role of biometrics in the conventional cryptosystem is just password management, as mentioned by Bruce Schneider.

2.3.1. Biometric Encryption: The Goal of a Biometric encryption is to embed secrecy into a biometric template in a way that can only be decrypted with a biometric sample from the enrolled person. Here Biometric Encryption is done by securely binding the key with the password in a database. When the biometric trait is presented live, the key retrieval algorithm generates the sequence of keys and Verification is done against the key stored in the database. The key is recreated only if the correct biometric live biometric sample is presented on verification. The key is randomly generated on enrollment, so that the user does not even know it [4, 5, 6, 7, 8]. “In Biometric Encryption, you can use the biometric to encrypt a PIN, a password, or an alphanumeric string for numerous applications – to gain access to computers, bank machines, to enter buildings, etc. The PINs can be 100s of digits in length; the length doesn’t matter because you don’t need to remember it. And most importantly,

all one has to store in a database is the biometrically encrypted PIN or password, not the biometric template." – As mentioned by Dr. George Tomko [8, 9, 10].

(1) Generating Pseudo Random Numbers: Cryptographic applications typically make use of algorithmic techniques for random number generation. These algorithms are deterministic and therefore, produce a sequence of numbers that are not statistically random. However, if the algorithm is good, the resulting sequences will overtake many reasonable tests of randomness. Such numbers are referred to as pseudo random numbers. Here we generate random numbers using the principle of chaos.[14]. The term chaotic is commonly used to describe a system that, although governed by a handful of non-linear equations, behaves in an apparently random manner. The main difference between chaos and randomness lies on the concept of determinism. As Random process cannot be predicted by any means, they are not deterministic and hence can't be used for key generation as we cannot get back the original sequence which would be required at the time of matching.

So the advantageous of chaos is that even very negligible differences in initial conditions would yield widely diverging outcomes for chaotic systems, rendering long-term prediction impossible. This happens even though these systems are deterministic, meaning that their future dynamics are fully determined by their initial conditions, with no random elements involved. In other words, the deterministic nature of these systems does not make them predictable. In biometrics, the biometric traits are unique to a particular individual and hence, there will be a unique value associated with everyone biometric, which will be the input value for generating the pseudo random numbers which would be the key for the biometric template.

If by some hook or crook, someone gets some numbers in the middle of the sequence, the resulting sequence would evolve very differently from the original which invariably would stop anyone from compromising the database. That is, Instead of the same pattern as before, it diverges from the pattern, ending up wildly different from the original. In biometric security, implementation is in hardware, so this chaotic number generator can be implemented in hardware very easily.

In this paper we generate Pseudo random numbers using the following and non linear equations. (1).Logistic Map (2). Tent Map. (3). Modified Logistic Map (4). Chinese Remainder Theorem.

2.3.2. Quadratic recurrence equation: The function we use to create pseudo random numbers that exhibit chaotic characteristics are: the logistic map, the tent map and modified logistic map. The logistic map is defined by a parabola, the tent map by a broken line, both symmetric about $X = \frac{1}{2}$. For both, the height of the maximum point is varied to define a family of functions. The height gives the family parameter.

First we generate pseudo random numbers with logistic map. A logistic function is a quadratic function of the form $X_{n+1} = rX_n(1-X_n)$, where r is a constant. The most interesting phenomena occurs as r varies in the range $2 < r \leq 4$. Here r is the catalyst for chaos.

It is a typical example of how complex, chaotic behaviour can arise from very simple non-linear dynamical equations. For a particular value of r, we may generate sequences $X_0, X_1, X_2, X_3, X_4, X_5, \ldots, X_m \ldots$ by choosing an initial value x0 and defining subsequent elements of the sequence iteratively by the rule $X_n + 1 = r X_n (1 - X_n) \ldots\ldots\ldots\ldots (1)$ The first few iterations of the logistic map give

$$X_1 = rX_0(1-X_0)$$
$$X_2 = r^2(1-X_0)X_0(1-rX_0+rX_0^2)$$
$$X_3 = r^3(1-X_0)X_0(1-rX_0+rx_0^2)*(1-r^2X_0+r^2X_0^2+r^3X_0^2-2r^3X_0^3+r^3X_0^4)$$

As r varies in the range $2 < u \leq 4$, the generic long term behaviour of sequences generated by the iteration changes dramatically. As r increases, convergence to a single limiting value is followed by convergence to a 2-cycle, then 4-cycle,8-cycle and cycles of higher powers of 2 and this behaviour continues until chaotic behaviour arises. Once chaotic behavaiour starts, no pattern is evident in the values produced by iteration.

These facts are well explained by the following bifurcation diagram which is obtained by plotting as a function of r, a series of values for X_n obtained by starting with a random value X_0 iterating many times, and discarding the first points corresponding to values before the iterates converge to the attractor. In other words, the set of fixed points of xn corresponding to a given value of r are plotted for values of r increasing to the right.

At r approximately 3.57 is the onset of chaos.. We can no longer see any oscillations. Slight variations in the initial population yield dramatically different results over time, a prime characteristic of chaos.

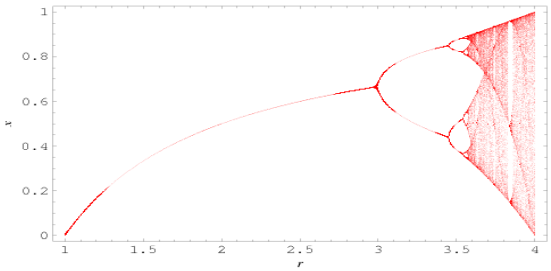

Figure.1. Bifurcation of logistic map

The above figure shows a bifurcation diagram of the quadratic recurrence equation which is obtained by plotting as a function of r series of values for X_n obtained by starting

with random value X_0 , iterating many times, and discarding the first points corresponding to values before the iterates converge to the attractor. In other words, the set of fixed points of X_n corresponding to a given value of r are plotted for values of r increasing to the right.

The Secret Key Stream Values are shown in Figure.2 and Figure.3 (Key Values are 0.23232300000000 and 0.89296), the bifurcation is obtained when we put r =3.541.

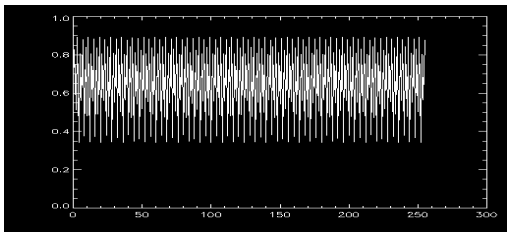

Figure.2. Logistic key stream

The probability density function of logistic is not uniform, but by introducing a proper threshold level, the output of the bit sequence becomes uniform. The control parameter and initial value of the map is determined. Then, a real value is generated by each iteration, which is converted into a bit by a single level threshold function. The threshold value is calculated using a computer simulation.

Figure.3. Secret key by using quadratic recurrence equation

(1) Algorithm: Let $b_i (i = 0,1,2,....)$ be the i^{th} output bit of the Logistic equation, which is generated according to the initial key, Key -P. $L-1$ integer pseudo random numbers. g_i' s (i =0,1,2,…. $L-1$) are calculated using these $b_i's$, as shown in the following equation

$$\begin{aligned} g_1 &= 1 \\ g_2 &= [(2b_0 + b_1)(1)/2^2 - 1)] + 1 \\ g_3 &= [(2b_2 + b_3)(2)/2^2 - 1)] + 1 \\ g_i &= [2^{j-1}b_k + 2^{j-2}b_{k+1} + ...+ b_{k+j-1})(i-1)/(2^j - 1)] + 1 \end{aligned} \quad (2)$$

Where $\lfloor \log_2 i \rfloor + 1$, $k = \sum_{s=2}^{i-1} \lfloor \log_2 s \rfloor + 1$, $\lfloor x \rfloor$ denotes the floor of x. since the number of permuted pixels is equal to the image size.(1). Get the Key Values from Biometric Trait, and then assign the values to variable A and B Respectively (2). Get the Biometric trait Size using the function of size () (3). Construct the loop using initialization parameter=0 followed by image size and then increment operator (4). Apply the quadratic recurrence equation and store the results into new array (5). Assign the new Array value to variable A (A=X) (6). Resultant New Array is sorted in ascending order Key Distribution Plot in IDL (I Plot) is

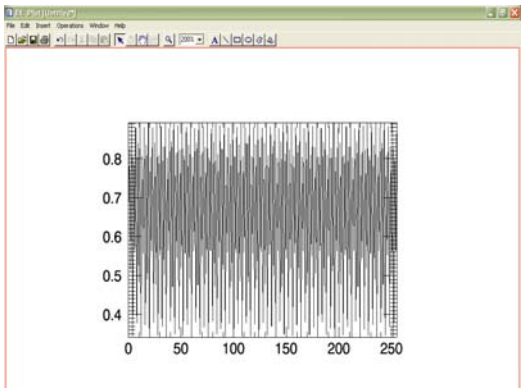

Figure.4. Logistic map key distribution

Figure.5. Key's Generated by IDL (Logistic Map)

2.3.3. Tent Map: The tent map (also called triangular map) function uses its previous output as present input. In this paper uses the following keys a=.7278346278462847, b=.3346462874623842

The tent map is an iterated function, in the shape of a tent , forming a discrete dynamical system. It takes a point X_n on the real line and maps it to another point. In nonlinear discrete dynamical systems the tent map, T: $[0,1] \to [0,1]$ defined by

$$f(x) = 1 - 2\left|x - 0.5\right| = \begin{cases} \mu x, 0 \le x \le \dfrac{1}{2} \\ \mu(1-x), \dfrac{1}{2} \le x \le 1 \end{cases} \quad (3)$$

Where $0 \le \mu \le 2$. The tent map is constructed by two string lines, which makes the analysis simpler than for truly nonlinear systems. The graph of the T function may be plotted by hand and is given by

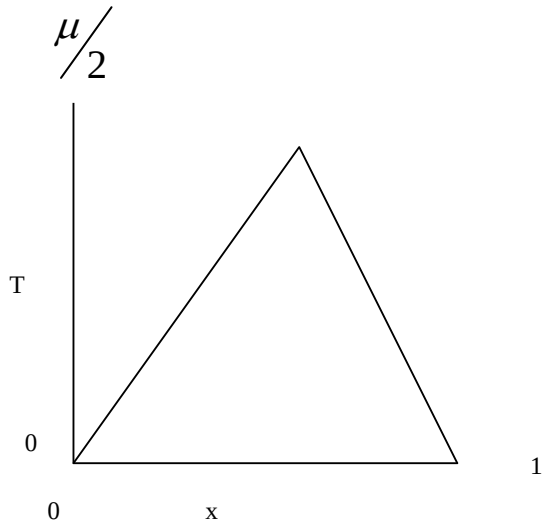

Figure .6.

The iterative map is $x_{n+1} = T(x_n)$ where $x_n \in [0,1]$. The Iteration of the tent map is will be

$$x(1) = \begin{cases} 2x(0) = x(0)1 = (0.b_2..b_j..b_{L-1}b_L)_2, & 0 \le x(0) < 0.5 \\ 2(1-x(0)) = (0.b_1'b_2'b_3'....b_j'..b_{L-1}'b_L')_2, & 0 \le x(0) \le 1 \end{cases} \quad (4)$$

Where $\{$ denotes the left bit-shifting operation. Note, that $b_1 = 0$ when $0 \le x(0) < 0.5$. Apparently, after L-1 iterations $x(L-1) \equiv (0.b_L)_2 = (0.1)_2$ Then $x(L) \equiv 1$, and $x(L+1) \equiv 0$. That is, the number of required iterations to converge to zero is $N_r = L+1$. Note that $N_r = 0$ when $x(0) = 0$. Algorithm: (1). Tent map is chosen as a chaotic system instead of a logistic map , since its probability density function, PDF, is uniform and implementation is almost simple. (2). Control parameter and initial condition of the map is determined by key-S. Each of them is defined with 64-bits and a simple linear transformation. (3). Real values of chaotic sequences are generated by iterations of the map: $x_0, x_1, x_2, ...x_{(nxn)}$ where n is the image size (4). 255 threshold levels in the range [0, 1] are defined and grey scales of pixels from 0 through 255 are attributed to them respectively. The Picture shows that the signals is random and non-periodic

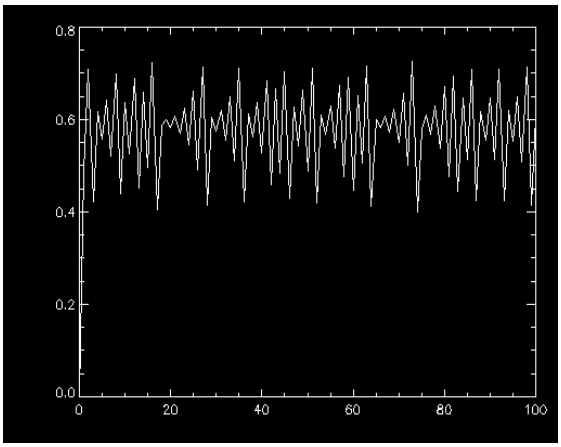

Figure.7. Tent Map (Implemented by IDL)

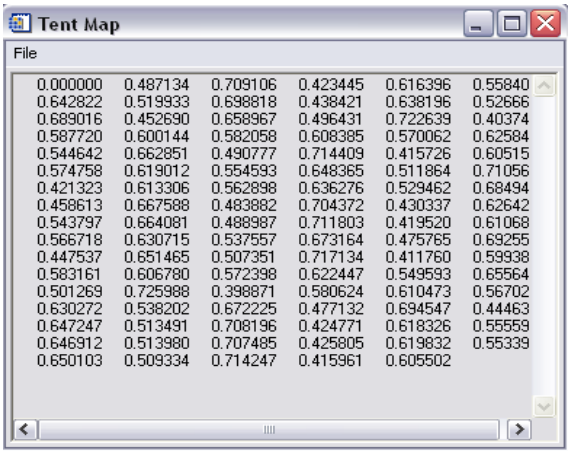

Figure.8. Tent Map Keys

Theoretical Analysis of Tent Map: In this Section, we consider the theoretical analysis of the runs in the pseudo-random numbers generated by the chaotic maps. In this analysis , we can understand that the distributions of runs generated by chaotic maps depend on the characteristics of the maps. The tent map is not symmetric with respect to the center a as shown in Figure.8.

Namely, the length of all run down is equal to be 1 and they are generated from an interval $[r1', 1]$. Moreover, after every run up ends, the rundown is generated without fail. Considering this feature, the probability of runs generated by the tent map with $a = 0.5$ can be expressed as

$$P_1 = \left(\frac{1}{2}\right) + \frac{1}{2}$$

.

.

.

$$P_d = \left(\frac{1}{2}\right)^{d+1} \tag{5}$$

The following figure shows the theoretical probability function of runs generated by the tent map, which is calculated by the equation of

$$P_d = \frac{1}{2}a^{d-1}(1-a) + \frac{1}{2}a(1-a)^{d-1} \tag{6}$$

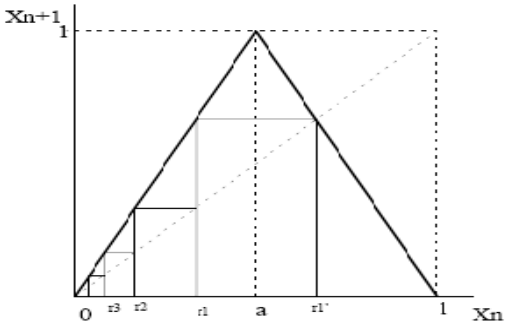

Figure.9. Run Test in Tent Map

2.3.4. Linear Congruential Generators: This algorithm is proposed by Lehmer which is known as the linear congruential method. The algorithm is parameterized with four numbers, as follows:

TABLE I.

M	the modulus	m > 0
A	the multiplier	0 < a < m
C	the increment	0 ≤c<m
X0	the starting value, or seed	$0 \le X_0 < m$

The sequence of random numbers X_n is obtained via the following iterative equation.

If m, a, c, and X_0 are integers, then this technique will produce sequence of integers with the integer in the range $0 < X_n < m$ .The Strength of the linear congruential algorithm is that if the multiplier and modulus are properly chosen, the resulting sequence of numbers will be statistically indistinguishable from a sequence drawn at random (but without replacement) from the set 1, 2, ……m 1. but there is nothing random at all about the algorithm , apart from the choice of the initial value X_0 . Once that value is chosen, the remaining numbers in the sequence follow deterministically.

Figure 10 Contains Pseudo Random Keys in IDL and Figure 11 is Key Distribution Plot

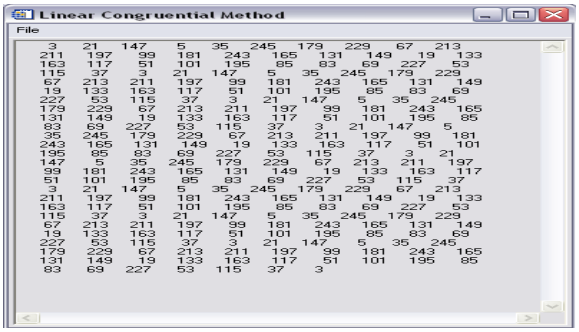

Figure.10. LCM Keys (IDL Output)

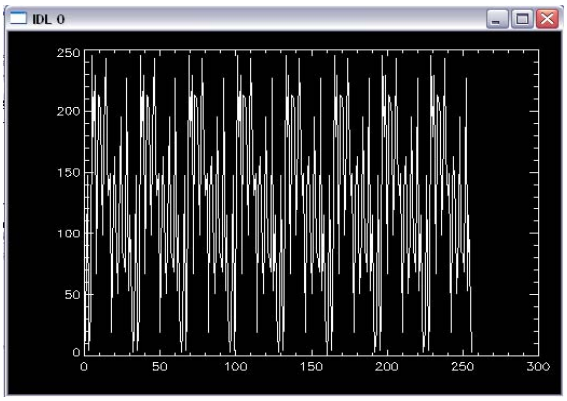

Figure.11. Key Sequence of LCM

2.3.5. Modified Logistic Equation: Pseudo Random numbers are generated by use a modified logistic map. The modified logistic map is one of the simplest chaotic maps. The map is expressed as the following equation

$$\begin{cases} X_{k+1} = \dfrac{2}{\alpha} X_k \left(1 - \dfrac{X_k}{2\alpha}\right) & 0 \le X_k \le \alpha \\ X_{k+1} = \left(\dfrac{X_{k+1} - 2\alpha}{2\alpha}\right)\left(2 - \dfrac{X_{k+1} - 2\alpha}{1-\alpha}\right) & (\alpha < X_k \le 1) \end{cases} \tag{7}$$

Where , α is the parameter changing the top of the map. Random sequences are like uniform random number.

This Modified Logistic map enhances the security and extra bifurcation parameter. The result of the M Logistic Equation (Figure.12)

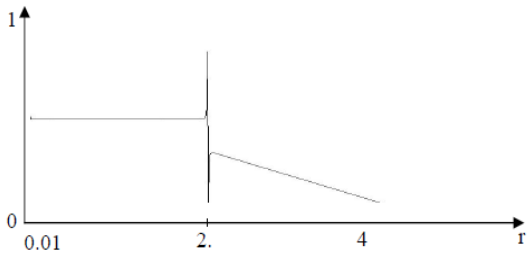

Figure.12. Bifurcation diagram of modified Logistic map for $0.01 \le r \le 4$

The Secret key stream values in Modified Map and Key Distribution plot in IDL is Shown in Figure.13 and Figure.14

Figure.13. M-Logistic Keys (IDL Output)

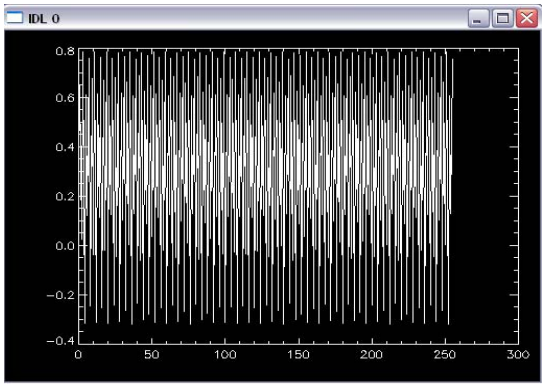

Figure.14. Key Sequence (IDL Output)

2.3.6. Encrypted Templates Based Enrollment & Verification Integrated Model: Any biometric authentication system can be viewed as a pattern recognition system. Such a system consists of biometric readers or sensors; feature extractors to compute salient attributes from the input signals; and feature matchers for comparing two sets of biometric features. An authentication system consists of two subsystems: one for enrollment and one for verification. During enrollment, biometric measurements are captured from a subject, relevant information from the raw measurements is gleaned by the feature extractor, and this information is stored in the database. During verification, that a person's biometric matches a claimed identity [4, 6, 11]. The system acquires the biometric sample from the subject, extracts features from the raw measurements, and searches the entire database for user acceptance.

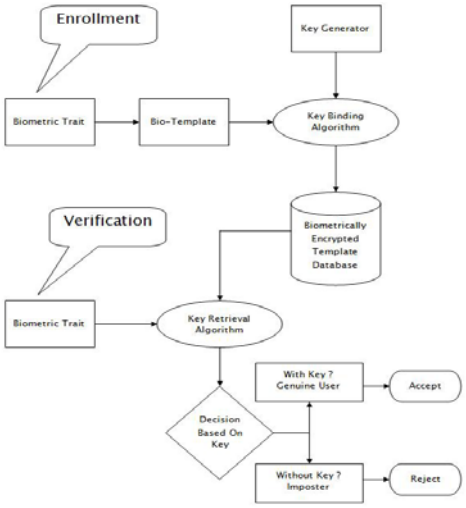

Figure.15. Data Flow Diagram of Key Based BE

In this case, an enrollment process consists of four major components like a biometric sensor, a key generator that normally outputs a random key, a binding algorithm that creates an encrypted template and database. A verification process consists of biometric sensor to capture a biometric sample, a key retrieval algorithm which applies the live biometric sample to the stored encrypted template in the database; after that retrieval algorithm brings the key if the biometric sample is genuine else user acceptance is denied [12, 13].

III.EXPERIMENTAL RESULTS

The proposed scheme is implemented in two different platforms; IDL and PHP-AJAX.A sequence of experiments was conducted to validate the effectiveness of the proposed scheme.

Key generated in this process is completely non-linear and there is no relationship between any two keys produced and as such hill climbing or prediction of data is no way possible.

In figure 16,17,18,19, 20 Live Bio-Trait is received by sensor, and then the key generator generates keys. Generated keys are validated against the stored biometric trait key. This works are done in both IDL and PHP-Ajax Platforms. This concept is implemented successfully in Biometric-based web access domain and will test the performance of the overall web access system. Ten files were created in a www root directory and Basic Authentication was used to restrict access to this

directory. Ten users were asked to evaluate the system. Seven out of the ten users were enrolled into the system. Each of the seven enrolled users was allowed to access a subset of the ten files. Over a period of three weeks, enrolled users accessed their files by providing their Fingerprint image each time. A user was accessing a set of files was not aware of the existence of the other files. The users were challenged to access other files or access the files without providing their Fingerprint but none of these attempts were successful. Access to the files could not be gained in any way other than providing genuine fingerprint images. Each of the enrolled user also tried to enter the system by impersonating the other six users, while the three users who were not enrolled tried to enter the system as one of the seven enrolled users. The Architecture of Biometric based web access is

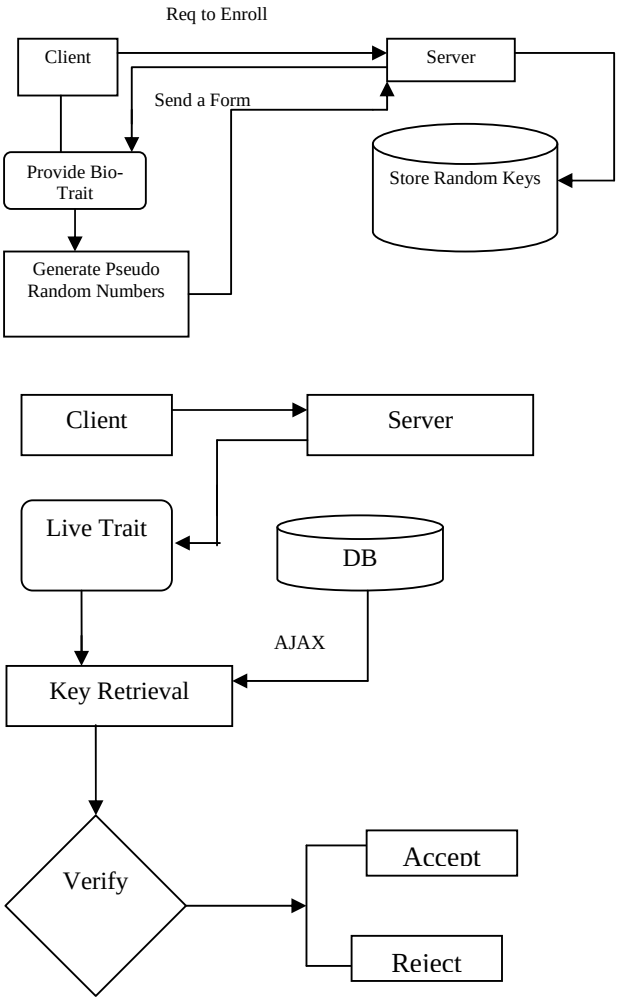

Figure.16. AJAX Technology in Biometric Security

Figure.12 [Ajax Technology is to reduce the post back operation in web domain and will increase the request and response process.]

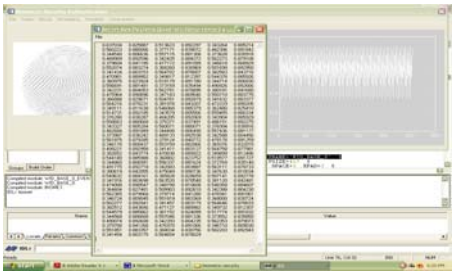

Figure.17. (IDL) Verification

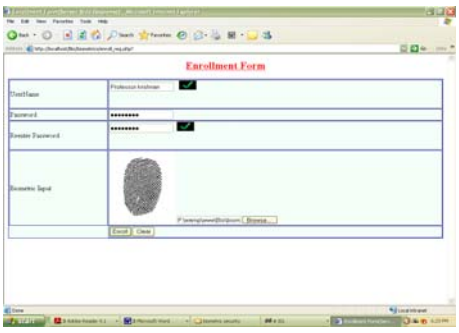

Figure.18. Enrollment Form

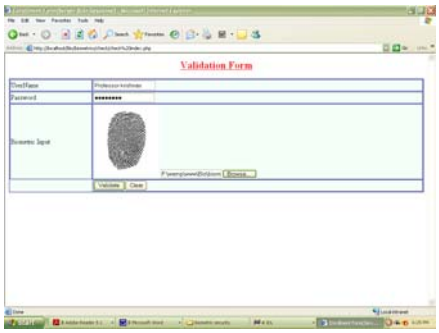

Figure.19.Verification Form (From Server Response)

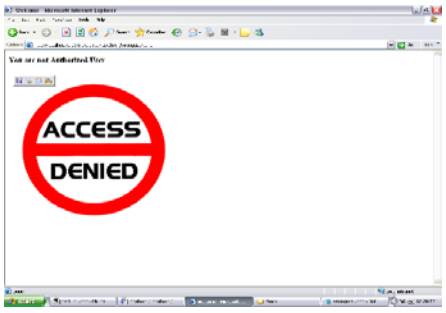

Figure.20. Unauthorized Access Output

IV.CONCLUSION

Here in this paper we proposed one authentication scheme to protect the biometric templates and to improve the security and privacy level of biometric authentication system. The main concept of the proposed authentication scheme is that we do not store any biometric trait in the database and verification process is done using the keys generated. The algorithm to generate the keys uses only the biometric traits that would be obtained from the user and the experimental results shows that the generated pseudo random numbers are so good that the numbers look exactly like there were really random i.e. numbers are non-periodic, non-repeating which eventually ensures very high security and privacy of the biometric authentication system.

Finally, we obtained the view of the security of our proposed authentication scheme against the attacks described in section 1. The performance of the authentication scheme is presented by the experiments and results.

REFERENCES

[1] Claus Vielhauer, "Biometric user authentication for IT Security from Fundamentals to Handwriting", 2006 Springer Science + Business Media, Inc.

[2] K.Jain, A.Ross, and S.Pankanti. "Biometrics: A Tool for Information Security". IEEE transactions on Information forensics and security , Vol .1 , No. 2, June 2006 , pp. 125-143

[3] The BioAPI Consortium, "BioAPI Specification Version 1.1", March 2001.

[4] U. Uludag, S. Pankanti, S. Prabhakar and A.K. Jain. "Biometric Cryptosystems: Issues and Challenges". Proceedings of the IEEE. 92(6):948-960. 2004.

[5] "Bruce Schneier, Applied Cryptography", 2nd Ed., John Wiley & Sons, Inc., New York, 1996.

[6] Ann Cavoukian and Alex Stoianov, "Biometric Encryption: A Positive-Sum Technology that Achieves Strong Authentication, Security and Privacy", March 2007.

[7] V. Bjorn. "Cryptographic key generation using biometric data". U.S. Patent 6035398, Mar. 7, 2000 (Priority date: Nov. 14, 1997).

[8] G.J. Tomko, C. Soutar, and G.J. Schmidt. "Biometric controlled key generation". U.S. Patent 5680460, Oct. 21, 1997 (Priority date: Sept. 7, 1994).

[9] G.J. Tomko and A. Stoianov. "Method and apparatus for securely handling a personal identification number or cryptographic key using biometric techniques". U.S. Patent 5712912, Jan. 27, 1998 (Priority date: July 28, 1995).

[10] G.J. Tomko. "Method and apparatus for securely handling data in a database of biometrics and associated data". U.S. Patent 5790668, Aug. 4, 1998 (Priority date: Dec. 19, 1995).

[11] Soutar, et al. "Biometric Encryption". In R.K. Nichols (ed.): ICSA Guide to Cryptography. McGraw-Hill. 1999.

[12] Soutar, D. Roberge, A.V. Stoianov, R. Gilroy, and B. V. K. Vijaya Kumar. "Method for secure key management using a biometric", U.S. Patent 6219794, Apr. 17, 2001 (Priority Date: Apr. 21, 1997).

[13] . Soutar, D. Roberge, A. Stoianov, R. Gilroy and B.V.K. Vijaya Kumar, "Biometric Encryption," ICSA Guide to Cryptography, McGrow-Hill, 1999, also available at http://www.bioscrypt.com/assets/Biometric_Encryption.pdf

[14] "Cryptography and Network Security Principles and Practices", Fourth Edition-William Stallings ,Page(227)

A.Senthil Arumugam received M.Sc. degree in Information Technology and E-Commerce from Manonmaniam Sundaranar University,Tirunelveli,India in 2003, M.Tech degree in Computer and Information Technology from Manonmaniam Sundaranar University, Tirunelveli, India in 2007 and M.Phil Degree in Computer Science from Manonmaniam Sundaranar University,Tirunelveli,India. Currently, he is the Ph.D Research Scholar of Centre for Information Technology and Engineering of Manonmaniam Sundaranar University,Tirunelveli,India. His research interests include Biometric Encryption and Image Processing, Cryptography, Open Source Software Development and Web Services. He is a Member of the IEEE.

Nallaperumal Krishnan received M.Sc. degree in Mathematics from Madurai Kamaraj University,Madurai, India in 1985, M.Tech degree in Computer and Information Sciences from Cochin University of Science and Technology, Kochi, India in 1988 and Ph.D. degree in Computer Science & Engineering from Manonmaniam Sundaranar University,Tirunelveli. Currently, he is the Professor and Head of Centre for Information Technology and Engineering of Manonmaniam Sundaranar University. His research interests include Signal and Image Processing, Remote Sensing, Visual Perception, and mathematical morphology fuzzy logic and pattern recognition. He has authored three books, edited 18 volumes and published 25 scientific papers in Journals. He is a Senior Member of the IEEE and chair of IEEE Madras Section SignalProcessing/Computational Intelligence / Computer Joint Societies Chapter.