IJCS International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
SKRGC Since 2012

http://www.ijcsjournal.com Volume 5, Issue 1, No 22, 2017 ISSN: 2348-6600
Reference ID: IJCS-261 PAGE NO: 1663-1672

Alagappa University, Karaikudi, India 15th -16th February 2017
IT Skills Show & International Conference on Advancements in Computing Resources (SSICACR-2017)
http://aisdau.in/ssicacr ssicacr2017@gmail.com

CONTINUOUS AND TRANSPARENT USER IDENTITY VERIFICATION FOR SECURE INTERNET SERVICES

R.JOTHI, M.Sc., M.Phil., M.KANCHANA, M.Sc.,
Asst.Prof. & Head, Department of Computer Application
Research scholar , Department of computer science,
Krishnasamy College of Science,Arts & Management For Women, Cuddalore.
josri7112011@gmail.com

Dr. C.BHUVANESWARI
Assistant professor & Head, Department of computer science,
Thiruvalluvar University College of arts and science,
Thiruvennainallur
Bhuvana.csdept@gmail.com

Abstract - Conventional verification systems, such as those controlling access to a secure room, do not usually require the user to re-authenticate him for continued access to the protected resource. This may not be sufficient for high security environment in which the protector resource name needs to be continuously monitored for unauthorized use. In such cases, continues verification is needed. The proposed approach for continues user verification is CASHMA (Context Aware Security by Hierarchical Multilevel Architecture) System. The protocol determines adaptive timeouts based on the quality, frequency and type of biometric data transparently acquired from the user. The new proposed system that continually verify the presence/participation of a logged in user. For most computer system, once the identity of the user has been verified at login, the system resources are typically made available to the user until the user exits the system. In such system username and password are used for authentication. Here biometric technique offer solution for secure and trusted authentication. We use multiple biometric verification for continuous and transparent verification. So that achieves Higher Security then traditional authentication system. To achieve high security over data transmission, AODV (ad-hoc on demand distance vector) protocol has been used in this proposed work. The proposed method provides security and fast transmission. Simulation result shows that, this proposed method minimizes the time delay and provides secure data delivery.

Index Terms—CASHMA, Security, Biometric User Verification.*(key words)*

I. Introduction

In order to reduce misuse of computers and produce high security over internet access, the proposed CASHMA system provides a secure platform for the user. It transparently acquires biometric data's from the user, then adaptively computes and refreshes session timeouts on the basis of the trust put into the system. Then continuously verifies the presence of a user working with a computer. If the verification fails, the system reacts by locking the computer and by delaying or freezing the user's processes.

This thesis presents a new approach for user verification and session management that is applied in the context aware security by hierarchical multilevel architectures (CASHMA) system for secure biometric authentication on the Internet. Our continuous authentication approach is grounded on transparent acquisition of biometric data with biometric encryption and on adaptive timeout management on the basis of the trust posed in the user and in the different subsystems used for authentication. The user session is open and secure despite possible idle activity of the user, while potential misuses are detected by continuously confirming the presence of the proper user.

II. Working Principle

The CASHMA authentication service includes (Figure .1) authentication server, computational servers, databases of templates and web services. Authentication server: an authentication server which, interacts with the clients.

Computational servers: a set of high-performing computational servers that perform comparisons of biometric data for verification of the enrolled users.

Databases of templates: that contain the biometric templates of the enrolled users (these are required for user authentication/verification).

Web Services: web services are various services that use the CASHMA authentication service and demand the authentication of enrolled users to the CASHMA authentication server. They have to be registered to the CASHMA authentication service, expressing also their trust threshold.

We use any kind of device like a smart phone or PC that contacts the online banking service , which replies the client to get an authentication certificate. Next process is once the CASHMA authentication server matches the biometric traits in a template database it creates the CASHMA certificate and transfers that certificate to the client and then client forwards the certificate to web service to get access.

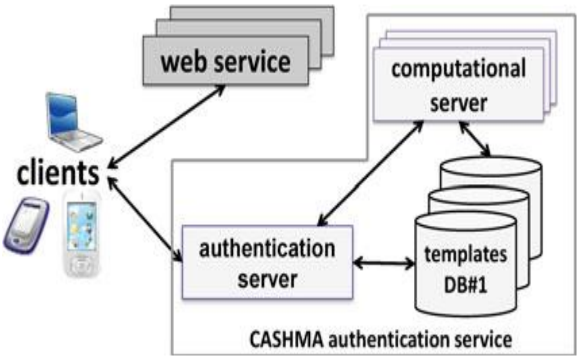

Figure.1: Overall view of the CASHMA architecture

III. CONTINUOUS AUTHENTICATION PROTOCOL

The continuous authentication protocol allows providing adaptive session timeouts to a web service to set up and maintain a secure session with a client. The timeout is adapted on the basis of the trust that the CASHMA authentication system puts in the biometric subsystems and in the user. The proposed protocol requires a sequential multi-modal biometric system composed of n uni-modal biometric sub-systems that are able to decide independently on the authenticity of a user. The main task of the proposed protocol is to create and then maintain the user session adjusting the session timeout on the basis of the confidence that the identity of the user in the system is genuine. The execution of the protocol is composed of two consecutive phases: the initial phase and the maintenance phase. The initial phase aims to authenticate the user into the system and establish the session with the web service. During the maintenance phase, the session timeout is adaptively updated when user identity verification is performed using fresh raw data provided by the client to the CASHMA authentication server. These two phases are detailed with the help of Figure. 2 and Figure. 3.

***Initial phase*:** This phase is structured as follows: For clarity, step 1 to step 4 are represented in Figure. 2 For the case of successful user verification only.

1. The user (the client) contacts the web service for a service request; the web service replies that a valid certificate from the CASHMA authentication service is required for authentication.

2. Using the CASHMA application, the client contacts the CASHMA authentication server. The first step consists in acquiring and sending at time t0 the data for the different biometric traits, specifically selected to perform a strong authentication procedure (step 1).

The application explicitly indicates to the user the biometric traits to be provided and possible retries.

3. The CASHMA authentication server analyzes the biometric data received and performs an authentication procedure. Two different possibilities arise here. If the user identity is not verified or additional biometric data are requested (back to step 1) until the minimum trust threshold gmin is reached. Instead if the user identity is successfully verified, the CASHMA authentication server authenticates the user, computes an initial timeout of length T0 for the user session, set the expiration time at T0 + t0, creates the CASHMA certificate and sends it to the client (step 2).

4. The client forwards the CASHMA certificate to the web service (step 3) coupling it with its request.

5. The web service reads the certificate and authorizes the client to use the requested service until time t0 + T0.

Maintenance phase: It is composed of three steps repeated iteratively:

1. When at time ti the client application acquires fresh (new) raw data (corresponding to one biometric trait), it communicates them to the CASHMA authentication server (step 5).

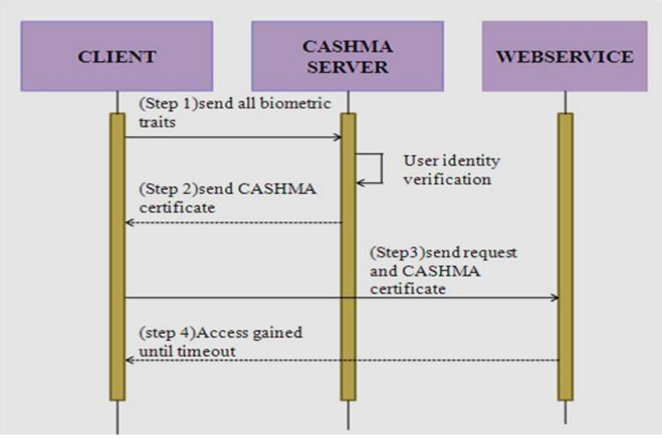
CLIENT
CASHMA SERVER
WEBSERVICE
(Step 1)send all biometric traits
User identity verification
(Step 2)send CASHMA certificate
(Step3)send request and CASHMA certificate
(step 4)Access gained until timeout

Figure 2: Initial phase in case of successful user authentication

The biometric data can be acquired transparently to the user; the user may however decide to provide biometric data which are unlikely acquired in a transparent way (e.g., finger-print). Finally when the session timeout is going to expire, the client may explicitly notify to the user that fresh biometric data are needed.

2. The CASHMA authentication server receives the biometric data from the client and verifies the identity of the user. If verification is not successful, the user is marked as not legitimate, and consequently the CASHMA authentication server does not operate to refresh the session timeout. This does not imply that the user is cut-off from the current session: if other biometric data are provided before the timeout expires, it is still possible to get a new certificate and refresh the timeout. If verification is successful, the CASHMA authentication server applies the algorithm to adaptively compute a new timeout of length Ti, the expiration time of the session at time Ti + ti and then it creates and sends a new certificate to the client (step 6).

3. The client receives the certificate and forwards it to the web service; the web service reads the certificate and sets the session timeout to expire at time ti + Ti (step 7).

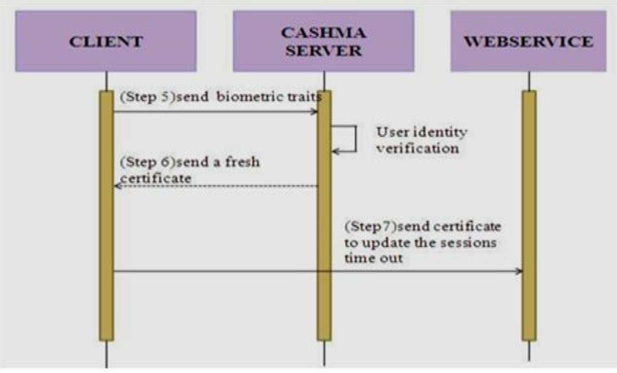
CLIENT
CASHMA SERVER
WEBSERVICE
(Step 5)send biometric traits
User identity verification
(Step 6)send a fresh certificate
(Step7)send certificate to update the sessions time out

Figure 3: Maintenance phase in case of successful user verification.

TRUST LEVELS

1. The subsystem trust level m (SK, t) is the probability that the uni-modal subsystem SK at time t does not authenticate an imposter (a non-legitimate user) considering both the quality of sensor (i.e., FMRk) and the risk that the subsystem is intruded.

2. The user trust level g (u, t) indicates the trust placed by the CASHMA authentication service in the user u at time t, i.e., the probability that the user u is a legitimate user just considering his behavior in terms of device utilization (e.g., time

since last key stroke or other action) and the time since last acquisition of biometric data.

3. The global trust level (u, t) describes the belief that at time t the user u in the system is actually a legitimate user, considering the combination of all subsystems trust level m(Sk=1..n.,t) and of the user trust level g(u, t).

4. The trust threshold gmin is a lower threshold on the global trust level required by a specific web service; if the resulting global trust level at time t is smaller than gmin (i.e., g (u, t) < gmin), the user u is not allowed to access to the service. Otherwise if g (u, t) > gmin the user u is authenticated and is granted access to the service.

TRUST LEVELS AND TIMEOUT COMPUTATION

CASHMA computes a new expiration time, when each time the CASHMA authentication server receives fresh biometric data from a user. Initial phase occurs at time t0 when biometric data is acquired and transmitted by the CASHMA system, and during the maintenance phase at time ti> t0 for any i = 1......, m new biometric data is acquired by the CASHMA system of the user.

A. Computation of Trust in the Subsystems:

Initially, the subsystem trust level could be simply set to static value m (Sk, t=1- FMR (Sk) for each uni-modal subsystem Sk and any time t (we assume that information on the subsystems used, including their FMRs, is contained in a repository accessible by the CASHMA authentication server).

In the initial phase m (Sk, t0) is set to 1 – FMR (Sk) for each subsystem Sk used. During the maintenance phase, a penalty function is associated to consecutive authentications performed using the same subsystem as follows:

penalty (x, h) = ex-h.

The calculation of the penalty is the first step for the computation of the subsystem trust level. If the same subsystem is used in consecutive authentications, the subsystem trust level is a multiplication of

- the sub system trust level m (Sk, ti_1) computed and

ii) the inverse of the penalty function (the higher is the penalty, the lower is the sub system trust level). m (Sk, ti) = m (Sk, ti_1). (Penalty(x, h))-1. Otherwise the subsystem is used for the first time or in non-consecutive user identity verification, m (Sk, ti) is set to 1-FMR (Sk).

B. Computation of Trust in the User:

During the maintenance phase, the user trust level is computed for each received fresh biometric data. The user trust level at time ti is given by:

g (ti) = (-arctan ((Δti – s). k) +π/2). Trust (ti-1) / -arctan (-s. k) + π/2

Value Δti =ti – ti-1 is the time interval between two data transmissions. Parameters k and s are used to tune the decreasing function: k impacts on the inclination towards the falling inflection point, while s translates the inflection point horizontally.

C. The Global Trust Level:

The global trust level is finally computed combining the user trust level with the subsystem trust level. In the initial phase, multiple subsystems may be used to perform an initial strong authentication. Let n be the number of different subsystems, the global trust level is first computed during the initial phase as follows:

$$\text{trust}\ (t0) = 1 - \Pi\ k=1....., n\ (1 - m\ (Sk, t0))$$

The global trust level in the maintenance phase is a linear combination of the user trust level and the subsystem trust level. The global trust level is computed using OR- rule. Results are,

$$\begin{aligned}\text{Trust}\ (ti) &= 1 - (1 - g\ (ti))\ (1 - m\ (Sk, ti))\\ &= g\ (ti) + m\ (Sk, ti) - g\ (ti)\ m\ (Sk, ti)\\ &= g\ (ti) + (1 - g\ (ti))\ m\ (Sk, ti)\end{aligned}$$

D. Computation of the Session Timeout:

The last step is the computation of the length Ti of the session Timeout. Starting from a given instant of time ti, we consider ti+1 as the instant of time at which the global trust level reaches the minimum threshold gmin, i.e., g (ti+1) = gmin. The timeout is then given by Ti = Δti = t i+1 - ti.

Equation (1), which allows the CASHMA service to dynamically compute the session timeout based on the current global trust level. It is then trivial to set the expiration time of the certificate at Ti + ti.

$$Ti = \{\tan\ (gmin.\ (\arctan\ (-s.k) - \pi/2\ /\ \text{trust}\ (ti) + \pi/2).\ 1/k + s$$

$$\text{If } Ti > 0 \text{ otherwise} \qquad (1).$$

IV.BIOMETRIC IDENTIFICATION

To overcome the limitations of password based authentication, later people use the biometric traits to achieve the strong authentication. The following are used as performance metrics for biometric systems:

False match rate (FMR, also called FAR = False Accept Rate): the probability that the system incorrectly matches the input pattern to a non-matching template in the database. It measures the percent of invalid inputs that are incorrectly accepted. *False non-match rate* (FNMR, also called FRR = False Reject Rate): the probability that the system fails to detect a match between the input pattern and a matching template in the database. It measures the percent of valid inputs that are incorrectly rejected.

Failure to enroll rate (FTE or FER): the rate at which attempts to create a template from an input is unsuccessful. This is most commonly caused by low quality inputs.

Receiver operating characteristic or Relative operating characteristic (ROC): The ROC plot is a visual characterization of the trade-off between the FMR and the FNMR. In general, the matching algorithm performs a decision based on a threshold that determines how close to a template the input needs to be for it to be considered a match.

If the threshold is reduced, there will be fewer false non-matches but more false accepts. Conversely, a higher threshold will reduce the FMR but increase the FNMR. A common variation is the Detection error trade-off (DET), which is obtained using normal deviation scales on both axes. This more linear graph illuminates the differences for higher performances (rarer errors).

Failure to capture rate (FTC): Within automatic systems, the probability that the system fails to detect a biometric input when presented correctly.

Template capacity: the maximum number of sets of data that can be stored in the system.

Equal error rate or crossover error rate (EER or CER): the rate at which both acceptance and rejection errors are equal. The value of the EER can be easily obtained from the ROC curve. The EER is a quick way to compare the accuracy of devices with different ROC curves. In general, the device with the lowest EER is the most accurate.

EFFICIENT FINGERPRINT ALGORITHM BASED ON TREE COMPARISON

We present a fingerprint matching algorithm that initially identifies the candidate common unique (minutiae) points in both the base and the input images using ratios of relative distances as the comparing function. A tree like structure is then drawing to connecting the common minutiae points from bottom up in both the base and input images. Matching score is obtained by comparing the similarity of the two tree structures based on a threshold value. Our proposed algorithm is also capable of comparing and producing matching scores between two images obtained from two different kinds of sensors. Our algorithm is sensor interoperable and also reduces the FNMR (false non-match rate) in cases where there is very little overlap region between the base and the input image.

Finding Common Points

The prime purpose of this phase is to find the number of common minutiae points available in a pair of fingerprint images. Given two fingerprint images with 'N1' and 'N2' identified minutiae points respectively (where N1 need not be equal to N2), this phase outputs the 'M' common minutiae points, which would be available in both the images. Effectively, if N1 represents the set of minutiae points in image 1 and N2 represents the set of minutiae points in image 2, M would be the intersection of N1 and N2 ($M = N1 \cap N2$).

We define a new term called the 'M (i) – tuple' to represent information about a minutiae that would identify it uniquely among the set of all minutiae. The M (i) – tuples of a pair of minutiae can be compared/matched to find if they both are the same or not.

When two images with identified minutiae points are given as input; the algorithm considers one image to be the base image (BM) and the other image to be the input image (IM). The areas marked in squares indicate minutiae points that are common in both the images, while the areas marked in circles indicate minutiae points that are unique to each of the two images.

M(I) – Tuples in base image (BM)

The base image has 'N1' number of minutiae points and N (BM) is the set of all minutiae in the base image. Now, the M (i) – tuple (i = 1 to N1) for each minutiae point is calculated as follows: For each minutiae i = 1 to N1, the 5 nearest minutiae points are found. This is done by calculating the Euclidean Distances from the 'i'th minutiae point to all the other minutiae points in the set N (BM) and noting down the 5 nearest minutiae points with respect to Euclidean Distances. Note that distance 'i – iN' means the Euclidean Distance between the points i and iN. So here, distance i – i1 means the Euclidean distance between minutiae point i and i1.

Matching phase:

In matching phase the algorithm does two functions. (1) Separates the Candidate Common Points List into two lists, (a) Confirmed Common Points List and (b) Spurious / Unconfirmed Point List.

(2) Uses the Confirmed Common Points List to generate a

Matching Score between the Base and the Input image.

Finding confirmed common points list

Which is the set of minutiae points in the base image, the algorithm considers only those points that feature in the Candidate Common Point List to create the tree. The remaining points in the set N(BM) are listed in the set N' (BM). After those points are considered, a tree like structure is drawn. Similarly from N(IM), algorithm considers points that feature in the Candidate Common Point List to create the tree and the remaining points are listed in the set N'(IM).

The lowest common point in both the images is considered to be the origin of an X –Y co-ordinate system. All the other points that are above this point are ordered with respect to their Y values (lower the Y value, lower the order, so the origin point is order 0, the next is order 1 and so on), and when two points happen to have the same Y value, the point with the lower X value is given the lower order.Effectively the order increases bottom up in the image. After ordering all the Candidate Common Points, they are connected from bottom up with respect to their order in both the images.

ENHANCED AND FAST FACE RECOGNITION BY HASHING ALGORITHM

For fast Face recognition, in this proposed work hashing technique has been used. The proposed technique employs the two Existing algorithms, 2-D discrete cosine transformation and K-means clustering. The images have been applied for different pre-processing phases and the above algorithms must be used in order to obtain the hash value of the face image.

Here binary search methods are used to increase searching process.The proposed technique is based on calculating numerical values of a face image for fast Recognition. These values should be unique for each face in the database. The Calculated hash value of the face is then saved into the database with the corresponding face ID. Storing the database with respect to hash values will help to apply modified binary search to recognize the faces from the database.

The whole process consists of two different phases: database generation phase and binary search for face recognition phase.

Database Generation Phase

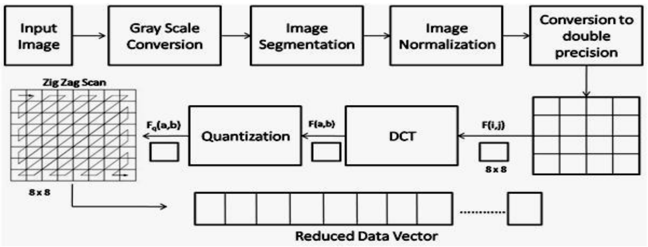

Figure 4: Pre-processing and dimensionality reduction

The input Images

A number of train face images were obtained for each individual to be stored in the database. Each train image for a person is different depending upon the conditions like pose, illumination, background and other facial expressions, among others. These factors will help to find the faces more accurately. All the images in the database should be in greyscale. Hence the very first step after image acquisition is to convert the image to grayscale.

Pre-processing

The pre-processing phase segments the face region from the input image. The face region is then normalized (scaling, sizing and positioning) in standard format of the stored database. The normalized image is then converted to double precision image intensity.

Data Reduction

The segmented faces are divided into many blocks in order to apply linear transformation to them. The block size may be 8×8 or 16×16 and lie in a sequence. Here we are using 2-D DCT which provides only N number of coefficients from the block of size MxM (N <MxM). For example M=8, i.e., block size be MxM which after applying DCT is represented by N, say N=8. These eight coefficients are high energy components in that block and have low frequency.

Face bases Generation

The DCT coefficients are then quantized to discard the non significant information from the image. These coefficients are then passing to the next step for Face base generation.

V.CONCLUSION

In this work, exploited the novel possibility introduced by multi modal biometrics to define a protocol for continuous authentication (CASHMA), that improves security and usability of user session. The protocol computes adaptive timeouts on the basis of the trust posed in the user activity and in the quality and kind of biometric data acquired transparently through monitoring in background the user’s actions.

VI. FUTURE ENHANCEMENT

Multimodal biometric systems elegantly address several of the problems present in uni-modal systems. By combining multiple sources of information, these systems improve matching performance, increase population coverage, deter spoofing, and facilitate indexing. Using finger vein, gait, voice and other biometric traits for verification significantly improve the performance of future works, with the widespread deployment of biometric systems in several civilian and government applications. Using advanced AODV protocols also increase the performance of the future work.

REFERENCES

[1] CASHMA-Context Aware Security by Hierarchical Multilevel Architectures, MIUR FIRB, 2005

[2] R. Brunelli and D. Falavigna, Person Identification using Multiple Cues, IEEE Transactions on PAMI, Vol 12, pp. 955-966, Oct. 1995.

[3]. N.K. Ratha, J.H. Connell, and R.M. Bolle, “An analysis of minutiae matching strength”, Proc. AVBPA 2001, Third International Conference on Audio- and Video-Based Biometric Person Authentication, pp. 223-228, 2001.

[4] John Daugman, Biometric Decision Landscapes, Technical Report, University of Cambridge, UK, 1999.

[5. D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, Springer, 2003.

[6 L. Hong and A. Jain, Integrating Faces and Fingerprints for Personal Identification, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 12, pp. 1295-1307, Dec. 1998.

[7] L. Hong, A. K. Jain, and S. Pankanti, Can Multibiometrics Improve Performance?, In Proceedings AutoID’99, NJ, USA, pp. 59-64, Oct. 1999.

[8] A. K. Jain, R. Bolle, and S. Pankanti, Multimodal Biometrics: Personal Identification in a Networked Society, Kluwer Academic Publishers, pp. 1-38, 1999.

[9] N. Poh and J. Korczak, Hybrid Biometric Authentication System Using Face and Voice Features, Third International Conference on AVBPA, pp. 348-353, 2001.