ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
**Indian Journal of Science and Technology,** *Vol 11(13), DOI: 10.17485/ijst/2018/v11i13/118273, April 2018*

# A Novel Approach for Enabling More Accurate Trust and Reputation Mechanisms with an Efficient and High-Security Remote Authentication in the Cloud Computing Environment

**Sabout Nagaraju[1] and S. K. V. Jayakumar[2]**

[1]Department of Computer Science, Pondicherry University Community College, Lawspet, Pondicherry – 605008, India; saboutnagaraju1983@gmail.com
[2]Department of Computer Science, School of Engineering & Technology, Pondicherry University, Kalapet, Pondicherry-605 014, India
skvjay@gmail.com

## Abstract

**Objectives:** To measure an accurate trustworthiness of dynamically changing usage patterns of the cloud computing necessitates the mathematical trust and reputation mechanisms along with the high-security remote authentication. **Method:** As a result, we have proposed an efficient trustee-based methodology that will provide trusted mutual authentication in order to protect sensitive information from the malicious users. The advantage of our scheme is that authentication key credentials and trust and reputation values are never revealed to the Cloud Service Providers (CSPs) and also assist stakeholders to select genuine, appropriate and trustworthy CSPs before registering any cloud services. **Findings:** In our proposed authentication approach, mutual identity will be verified based on n-party bilinear pairing key distribution and random nonce. Trust and reputation values are calculated using direct evidence, indirect evidence and user's historical feedbacks. In addition to formulating an effective authentication and an accurate trust and reputation mechanisms, we have analyzed completeness of proposed approaches using Gong, Needham and Yahalom (GNY) cryptographic belief logic and different data sets. **Application/Improvements:** The performance analysis shows that proposed framework resist from various impersonation attacks and gives better computational efficiency and also results better in fingerprint samples recognition rates than existing schemes.

**Keywords:** Authentication, Privacy, Reputation, Trustee

## 1. Introduction

In the present era of Information and Communications Technology (ICT) world cloud computing is a predominant paradigm that delivers economical data storage and high-speed computing capabilities for stakeholders. It promises the stakeholders to make their daily life smarter with new technologies such as Internet of Everything (IOE) and advanced mobile technologies. In fact these new imperatives are rapidly emerging as central pillar for digital urbanism.[1] As per Gartner predict report 2015 the public cloud services expected growth by the year 2016 is $210 billion.[2] As per ABI research report, revenues of $5.2 billion are driven by the cloud computing through 240 million mobile business customers.[3] On the other hand cloud computing framework acts as a big black box as nothing is known to the users about what happens inside.[4] Clients have no control over data storage, processing and services.[5] Moreover, consumer's data will co-reside with more than one consumer, so there is no guarantee of security of things. Cloud computing is still subject to traditional data security attacks as well as authenticity, trust and reputation management concerns. To preserve authenticity, trust and reputation properties in cloud is given higher priority as these help to gain the clients control over the cloud services.[6]

*Author for correspondence

## 1.1 The Problems

In the present cloud computing framework, the following two key concerns have to be resolved. These two concerns not only prevent the legitimated users to access trusted cloud services from the authenticated CSPs, but also creates fear of loss of control over data and applications in the cloud.[5]

### *1.1.1 High-Security Authentication*

In the presented article, authors discussed and analysed the cloud-based authentication problems and risks from different perspectives.[5-11] In the recent years, several authentication approaches have been proposed in.[12-20] Each presents a partial view of authentication and lacking in providing an efficient and high-security authentication. There are three major problems needed to be solved. First, secure session keys generation and establishment, since the public cloud computing environment is insecure. An adversary can easily work out on session keys cipher-text because keys produce very small amount of cipher-text. Due to this reason there is a chance of leak age of authentication credentials. Second, single sign-on authentication requirement, since the users may access various services from the multiple CSPs. However, most of the single sign-on authentication approaches require a trustee participation and trustee could become a bottleneck for the user authentication. Third, the computation cost need to be minimized, since the cloud computing emerges with smart phones which have limited computing capacity.

### *1.1.2 Trust Calculation and Management*

As discussed in the reported articles cloud service providers may not be trustworthy so that the security of the sensitive information will be at risk.[21-23] For some financial gain, dishonest CSPs/staff may steal the user authentication credentials for acquiring sensitive information. Similarly, for maintaining reputation, cloud service providers may hide the data loss; instead they may return the stale data. CSPs may completely deny the service requests or execute few requests in order to save computing resources. Many trust and reputation management mechanisms have been proposedin.[24-28] Each presents a partial view of trust and reputation which lack in providing more accurate mathematical formal mechanisms.

As per our literature study existing research works have not taken effort to enable more accurate mathematical formal trust and reputation mechanisms with an efficient and high-security remote authentication protocol. To fill up this gap, in this article a trusted authentication framework is proposed. This framework helps the stakeholders to select genuine, appropriate and trustworthy cloud service providers before registration of some cloud computing services. With this critical new integration, stakeholders can maintain the control over cloud services. This shows the novelty and its scientific impact on the present era of cloud computing.

## 1.2 Our Research Contribution

The proposed cloud-based trusted authentication framework has the following main contributions.

1) An accurate global trust evaluation and management: In contrast to existing mechanisms, we have proposed a more accurate mathematical formal global trust evaluation mechanism. In this approach we have combined direct evidence, indirect evidence and user's historical feedbacks to accurately compute CSPs trust and reputation values. This approach can resists from fraudulent CSPs/staff as well as enforces guarantee of SLAs and PLAs.
2) Distributed trustee: In our investigation, a distributed trustee module is proposed for computing and maintaining the trust and reputation values as well as to store and provide the credential parameters for user identification. Trustee module first collects the genuine SLA and PLA copies from the users and CPS's. Next, collects and audits the genuine feedbacks and then computes the trust and reputation values from historical feedbacks, direct and indirect evidences available. Finally, these values are updated in the highly secured and distributed databases and also reply these values if any users request.
3) Selection of trustworthy CSPs: We have proposed an algorithm to choosegenuine, appropriate and trustworthy CSPs based on various service attributes and Global Trust Value (GTV). Two case studies are presented to show that the proposed algorithm selects highly trustworthy CSPs based on assigning random weights to the cost, trust and reputation values.
4) Trulysecure session keys generation and establishment: In the process to develop an efficient and high-security trusted authentication protocol, we have generated and established one-time session keys based on an enhanced n-party Diffie-Hellman bilinear pairing key distribution. Innovations done so far in cloud-based

authentication have not concentrated on this type of key distribution adoption. This approach overcomes many session key management problems in the cloud.

5) Time-based dynamic nonce: Based on system time we have generated unique dynamic nonce which is used for handshaking and protecting an alteration of communication messages. This approach helps to avoid various impersonations and replay attacks.
6) Strong Fingerprint data Privacy: The fingerprint data of every legitimated stakeholder is protected from the dishonest CSPs/staff. In the proposed authentication protocol, CSP servers are just allowed to match the hashed authentication parameters as well as fingerprint details are not stored in cloud servers. So that the malicious CSP/staff cannot steal the fingerprint data.
7) Provable Security: Formally we have analysed the completeness and security strength of the proposed protocol using very famous GNY cryptographic logic and four benchmark databases.
8) Computational Efficiency: The proposed protocol consumes eight bilinear pairings and six hashing operations for credentials validation (i.e., $O(8T_{bp}+6T_h)$). An existing scheme performsten bilinear pairings, eleven hashing, six multiplication and three division operations (i.e., $O(10T_{bp}+11T_h+6T_m+3T_d)$).

This paper further divided into seven sections. Literature reviews are presented in Section 2. System-level model and assumptions of our trusted authentication framework is illustrated in Section 3. Section 4 provides system preliminaries. Section 5 describes our investigation. Section 6 discusses completeness of the proposed authentication protocol. Section 7 reports the security and performance evaluation. Section 8 summarizes the proposed trusted authentication framework.

## 2. Related Work

Developing an efficient, robust and more convenient trusted authentication mechanism for the distributed cloud computing environment is a challenging research problem. In the recent years, several cloud-based authentication mechanisms have been presented in.[14-22] Fingerprint-based authentication received lot of attention in cloud computing to protect the unauthorized access of sensitive and personal information from the malicious users. Large number of in-house applications is using fingerprint-based authentications.[23] Because of accuracy, reliability, convenient and high security strength, fingerprint-based authentication is universally accepted for any kind of operational environments.[24] A. J. Choudhary *et al.*[14] described a lightweight mutual authentication scheme by incorporating user ID and password into smart cards with a proper key management aiming to provide user friendly protocol. However this protocol will work out well for only private cloud computing. In[15], J. Yang *et al*. presented a biometric fingerprint based mechanism for protecting cloud computing communications using geometric and Zernike moments. This scheme is lacking in providing robust and reliable authentication process as well as stores fingerprint details in cloud servers which are unique and once compromised, these details cannot change over the time. In[16], authors discovered an authentication mech based on decentralized access control policies. In this approach CSP know search record access policy. So, it is not suitable for protecting personal and sensitive information from the malicious CSP. In[17], Neil Zhenqiang Gong et al. designed a secure trustee-based probabilistic model and defense methods to identify the forest fire attacks and their preventions. This social authentication scheme is suitable for social network systems such as face-book and twitter, and it is not suitable for protecting sensitive information related systems such as e-governance and e-health care described a privacy preserved authentication scheme using digital signatures for securing electronic medical records. This scheme cannot resist the masquerade attack, because if a user lost or shared his/her certificate with any other registered users then they can authenticate successfully instead of a legitimate user.

In, authors designed a privacy-preserving authentication mechanism for protecting user privacy from the malicious cloud servers in data sharing applications. Proposed protocol provides an anonymous access request matching, access control at attribute level and proxy re-encryption for collaborative multi-user cloud computing services. Jun Zhou et al. established a privacy preserved authentication model for cloud based m-Healthcare systems to facilitate an efficient patient treatment from the direct and indirect authorized and un authorized physicians using attribute-based access tree. The mechanisms of are vulnerable to malicious service provider/ dishonest insiders. In literatures, authors described trustee-based biometric fingerprint authentication schemes for cloud computing environment to facilitate mutual authentication and secure key management for cloud users. Observing limitations, these two schemes

perform computational overhead and cannot resist from collusion, white-washing and good and bad mouthing attacks.

As per the above literature study, existing authentication mechanisms have not taken effort to provide an efficient and robust authentication protocol and are lacking in enabling trust and reputation management with the user registration and authentication process.

In the recent years, very few substantial trust and reputation calculation mechanisms have been proposed for cloud computing to find out the trustworthiness of collaborative cloud service providers. In, authors presented a trusted framework using technical and policy-based approaches. In this investigation, trustee related key issues and challenges were analyzed using detective controls. The demerit of this scheme is that author's main focused on logs accountability of the five kinds of abstraction layers. But, still there are number of factors like cloud data security, privacy, data availability, data transmission and reputation need to be considered for an efficient trust evaluation developed a framework for reliable file exchange based on request-response information. In, authors discovered a methodology to evaluate trustworthiness of CSPs by using armor to constantly monitor. The approaches of are not suitable for distributed public cloud computing because of communication overhead and unrealistic assumptions. In, authors presented a scheme for assessing global trustworthiness of CSPs based on security, reliability and availability attributes. This investigation proposed a trust-based SLA management model at client side to enforce SLA guarantee. addressed some limitations in the existing trust mechanisms and proposed more rigorous mechanisms based on collected evidence and attribute certifications. In enabled mutual trust among data owners, authorized users and CSPs through auditing log data received from the virtual machines. Existing approaches in literatures are lacking in providing a formal framework.

## 3. System-Level Framework and Assumptions

In this section a system-level framework is presented for distributed cloud computing environment which consists of data owner, cloud service providers, distributed trustee and authorized users as shown in Figure 1. The personal and sensitive information of a data owner or enterprise will be managed in the geographically distributed cloud data centers. The cloud service providers outsource cheap, flexible and on-demand storage space and computing capabilities to the data owner to make this information available any time to the legitimated users. Trustee is an organization that has capabilities to collect, store, compute and manage the authentication parameters, and trust and reputation values. Trustee services are distributed geographically with shared and highly secured databases. In this framework, data owners, legitimated users and CSPs must rely on distributed trustee. The notations and their meanings we used for describing our framework are listed in Table 1.

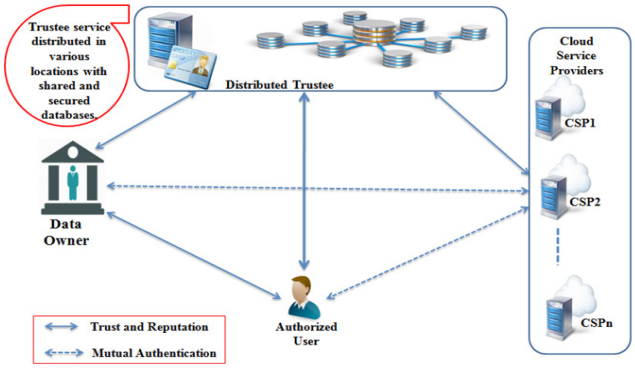


**Figure 1.** Trusted authentication model for cloud computing environment.

**Table 1.** Main notation definitions

| Notations | Meaning |
|---|---|
| $U_i$&$CSP_j$ | User i and Cloud Service Provider j |
| $E_{PB}$ (.) | A public-key encryption function |
| $D_{PR}$(.) | A decryption function's corresponding to $E_{PB}(\cdot)$ |
| $e_K$(.) | A symmetric encryption's function |
| $d_K$(.) | A symmetric decryption's function corresponding $e_K(\cdot)$ |
| Ê | Exponential function |
| UID* | The ID which $U_i$ inputs in authentication phase |
| PWD* | The password which $U_i$ inputs in authentication phase |
| BF* | The bio-metric fingerprint which $U_i$ submits in authentication phase |
| UID | The ID which $U_i$ inputs in registration phase |
| PWD | The password which $U_i$ inputs in registration phase |
| BF | The bio-metric fingerprint which $U_i$ submits in registration phase |

*(Continued)*

| Notations | Meaning |
|---|---|
| $C_i$ | The cipher text sent by various communication entities. |
| $T_{bp}$ | Bilinear pairing operation time |
| $T_m$ | Multiplication operation time |
| $T_d$&$T_c$ | Division operation time and Concatenation operation time |
| $T_X$&$T_h$ | Ex-OR operation time and One-way Hash operation time |
| $T_{dp}$&$T_{priv}$ | Trust on data processing and Trust on data privacy |
| $R_{val}$&$GT_{val}$ | Reputation value and Global Trust value |
| α, β and ã | Importance given to service cost, trust and reputation values |
| $T_{dt}$&$ST_{val}$ | Trust on data transmission and Service Trust value |
| K & $K_i$ | Shared session key and Shared session key of communication entity i |
| ST & DT | Cloud Service Type and User Data Type |
| SS & DS | Cloud Storage Size and User Data Size |
| PS & RS | Cloud Processing Speed and User Requested Speed |
| SC & SP | Cloud Service Cost and User Service Pay |
| $C_d$ | Cost Difference (i.e., $C_d$=SC – SP) |
| AT&RAT | Authentication Type and Requested Authentication Type |
| $C_{dminaccval}$ | Minimum Acceptable Value of Cost Difference |
| $ST_{minaccval}$ | Minimum Acceptable Value of Service Trust |
| $GT_{minaccval}$ | Minimum Acceptable Value of Global Trust |
| SPID | The Service Provider ID which $CSP_j$ inputs in the registration phase |
| SPID* | The Service Provider ID which $CSP_j$ inputs in the authentication phase |
| $h_i(.)$ | $i^{th}$ one-way hash function |
| \|\|& ⊕ | Concatenation operation and X-OR operation |
| $T(u, s)^t$ | User u has the trust in service type s at current time t |
| Salt | A random data that is used in generating a hashed password and also avoids the hash collisions. |
| +K & –K | Public keys and Private keys |
| ERN | Encrypted Random Number |
| FPR&FNR | False Positive Rate and False Negative Rate |

## 4. System Preliminaries

The bilinear pairing and n-party key agreement related preliminaries we used for designing and developing our mathematical formal framework are described in this section. Let $G_1$, $G_2$, $G_3$ be three cyclic additively-written groups and let $G_T$ be acyclic multiplicatively-written groups of an exponential base g with a large prime number order p.

**Definition 4.1.** Let a mapping $\hat{e} = G_1 \times G_2 \times G_3 \rightarrow G_T$ is a bilinear pairing that has characteristics as follow:

1) Bilinearity: $\forall$ a, b, c $\epsilon F_q^*$, $\forall$ g $\epsilon$ ($G_1$, $G_2$, $G_3$), ê ($g^a$, $g^b$, $g^c$) = ê(g, g, g)$^{abc}$.
2) Computability: Bilinear groups and bilinear mapping are computed efficiently.
3) If ê(g, g,g)=1, then bilinear pairing preserves non-degeneracy property.

**Definition 4.2.** Let êbe a bilinear pairing on (G1, G2, G3). The bilinear diffie-Hellman pairing for $\forall$ a,b, c $\epsilon F_q^*$, $\forall$ g $\epsilon$($G_1$, $G_2$,$G_3$) can be computed as ê($g^a$, $g^b$, $g^c$)=ê(g,g,g)$^{abc}$.

The above definitions and properties we used in our authentication process for establishing and generating shared session keys among the users, cloud service providers and trustee. In key generation process there are up-flow and down-flow stages. In up-flow stage, each entity computes intermediate secrete values and in the down-flow, intermediate results will be sent to the communication entity group to generate shared session keys. Where the communication entities involved in authentication process are denoted as $E_1, E_2 \ldots E_n$. Trustee chooses an exponential base α and a large prime number p as an order and secretly shares these values to the authorized users and CSPs.

During up-flow, each communication entity $E_i$ performs single exponent and concatenates resultant value to the received intermediate values as given in equation (1) and then sends it to $E_{i+1}$.

$$E_i \longrightarrow E_{i+1}\, \alpha^{\Pi(N_K \mid K \in [i,j])} \mid j \in [1,i] \qquad (1)$$

The up-flow process ends and the down-flow process starts when $E_i = E_n$,$E_{i+1}$ receives the up-flow as formulated in equation (2)

$$E_i \longrightarrow E_{i+1}\, (\alpha^{N_1}, \alpha^{N_1 N_2}, \ldots, \alpha^{N_1 N_2 \ldots N_i}) \qquad (2)$$

Upon receipt of the resultant flow, $E_n$ computes the shared session key K as given in equation (3) by exponentiation of secrete value $N_n$ chosen by $E_n$.

$$K = K_n = (\alpha^{N_1}, \alpha^{N_1 N_2}, \ldots, \alpha^{N_1 N_2 \ldots N_i})^{N_n} \quad (3)$$

$$E_{casp_{bk}}(BASID \| h(PWD) \| v)$$

Once the shared session key $K_n$ is computed, $E_n$ starts the down-flow with n-1 intermediate values as formulated in equation (4)

$$(\alpha^{N_1 N_n}, \alpha^{N_1 N_2 N_n}, \ldots, \alpha^{N_1 N_2 \ldots N_{n-2} N_n}) \quad (4)$$

Upon receipt ofn-1 intermediate values, each entity $E_i$ computes the shared session key as given in equation (5)

$$K = K_i = (\alpha^{N_1 N_2 \ldots N_{i-1} N_{i+1} \ldots N_n})^{N_i} \quad (5)$$

The down-flow ends when $E_i = E_1$.

# 5. Trusted Authentication Protocol

This section describes a trusted authentication protocol which will relate the trust and reputation management with user authentication. In this protocol, authentication parameters' verification will be performed in the cloud. The credential parameters and cloud service cost, trust and reputation values will be stored and managed in the distributed trustee servers. This approach helps the users to protect their authentication credentials and sensitive information from the dishonest CSPs and unauthorized adversaries.

## 5.1 Authentication Protocol

The protocol has three phases as follow.

### *5.1.1 Initialization Phase*

First, trustee chooses a random value and considers it as private key ($PR_T$) and then computes its corresponding public key as $PB_T = h_1(PR_T)$, where $h_1$is a one-way hashing function. Next, chooses values for g and p of the bilinear pairing function. Similarly, CSPs chooses a random values as private key ($PR_{CSP}$) and computes its corresponding public key as $PB_{CSP} = h_2(PR_{CSP})$, where $h_2$is a one-way hashing function. Trustee and CSPs publishes their public keys for cloud stakeholders.

### *5.1.2 Registration Phase*

Cloud user registration with the trustee will be performed by using following steps.

1) Each User ($U_i$) filters CSPs based on service attributes like {ST∋DT, SS≥DS, PS≥RS, SC≤SP, AT⊇RAT} and then sends a request to the trustee to get selected CSPs trust and reputation values.
2) User chooses a $CSP_j$ by using Algorithm 2 and sends his/her chosen $CSP_j$, user-id(UID), password (PWD) and Bio-metric Fingerprint data (BF) to the trustee for registration by encrypting with $PB_T$.
3) Trustee decrypts the user details and computes $h_3$(PWD+salt)=HBF, $h_3(\delta_{RN}(BF)) = h_3(RN \oplus BF)$, and $\mathbf{e}_{BF}(RN)$ =ERN, where $h_3(.)$ is the one-way hashing functions, $\mathbf{e}_{BF}(.)$ is the symmetric encryption function using BF and stores these values in distributed and highly secured databases.
4) Trustee sends UID, g, p and mutual operation will be performed on nonce to $U_i$ and $CSP_j$ through secure channel.

Assumption: CSPs are registered entities with the Trustee.

### *5.1.3 The Authentication Phase*

Performs following steps to validate $U_i$ authentication credentials.

1) User $U_i$ inputs UID*, chooses a random secrete number a (a<p) and nonce $n_1$ and then computes $X_A = g^a \bmod p$ and performs public key encryption on concatenation of UID*, $X_a$ and $n_1$ as C1= $E_{PB_{CSP}}(UID^* \| X_a \| n_1)$.$U_i$ sends $C_1$ to $CSP_j$.
2) $CSP_j$ obtains $U_i$ message details such as UID*,$X_a$ and$n_1$ by decrypting $C_1$ using private key $PR_{CSP}$. If UID*==UID, then $CSP_j$ selects a random secrete number b (b<p) and calculates $X_b = g^b \bmod p$, $X_{ab} = \hat{e}(X_a, X_b) \bmod p$, $n_2 = n_1 + 1$ and then derives $C_2 = E_{PB_T}(SPID^* \| UID^* \| X_a \| X_{ab} \| n_2)$ using trustee public key $PB_T$. $CSP_j$ sends C2 to trustee. If UID* is not matched, then the user request will be rejected.
3) Trustee obtains $CSP_j$ message details such as SPID*, UID*, $X_{ab}$and $n_2$by decrypting $C_2$ using private key $PR_T$. If SPID*==SPID and UID*==UID and are valid then trustee chooses a secrete number c (c<p) and computes session key $K = K_j = X_{ab}{}^c \bmod p$, $X_c = g^c \bmod p$, $X_{bc} = \hat{e}(X_b, X_c)$

mod p, andderives$C_3=(e_K(ERN||n_3)||X_{ab}||X_{bc})$. Trustee sends $C_3$ to $U_i$. If UID*or SPID* is not found with trustee, then the authentication request will be rejected.

4) From$C_3$, $U_i$ computes session key as $K=X_{bc}^{a}$mod p and obtains encrypted random number ERN and nonce $n_3$by using K and then checks $n_3$with $n_1+2$, if it is equal, then user inputs PWD* and BF*. $U_i$ decrypts random number using BF* as $d_{BF*}$ (ERN) = RN and then computes hashed password i.e., $h_3$(PWD*+salt), hashed fingerprint data i.e., $h_3$(RN⊕BF*)and derives a message as $C_4=e_K(HPWD*||HBF*||n_4)$. $U_i$ sends $C_4$to the trustee. If $n_3$is not equal to$n_1+2$, then the authentication process will be terminated.
5) Trustee decrypts $U_i$ sent message details such as HPWD*, HBF* and $n_4$using session key K and then checks $n_4$value with $n_3+1$, if it is equals, then computes intermediate secrete as $X_{ac}=\hat{e}(X_a, X_c)$ mod p and derives authentication credentials message as $C_5=e_K(HPWD*||$ HPWD$||HBF*||$HBF$||X_a||X_{ac}||n_5)$. Trustee sends $C_5$to $CSP_j$. If $n_4$is not equal to$n_3+1$, then the authentication process will be rejected.
6) $CSP_j$ computes session key using intermediate secrete $X_{ac}$ as K= $X_{ac}^{b}$modp and then decrypts authentication parameters such as HPWD*, HPWD, HBF*, HBF and $n_5$ using K. $CSP_j$ matches user submitted credentials with the registered parameters if ((HPWD*==HPWD&&HBF*==HBF)&&($n_5$==$n_2$+3)), then service will be provided to the user, otherwise, the authentication request will be rejected.

The user authentication process with CSP is described in Algorithm 1:

### *Algorithm 1: Authentication Phase*

Input: User-ID, password, biometrics fingerprint data and random numbers.

Output: Status of authentication process.

1) $U_i$ Inputs UID* and selectsa random secrete a (a<p) and nonce $n_1$
   Computes $X_A=g^a$mod p
   Derives $C_1= E_{PB_{CSP}}(UID*||X_a||n_1)$
   $U_i \xrightarrow{C_1} CSP_j$
2) $CSP_j$ Decrypts $C_1$ as $D_{PR_{CSP}}(C_1) = (UID*||X_a||n_1)$
   if UID*==UID, then $CSP_j$selects a random secrete b (b<p) and calculates $X_b = g^b$ mod p, $X_{ab}=\hat{e}(X_a, X_b)$ mod p, $n_2 = n_1+1$
   Derives $C_2= E_{PB_T}(SPID*||UID*||X_a||X_{ab}||n_2)$
   $CSP_j \xrightarrow{C_2}$Trustee
   If UID* is not found or invalid, then user request will be rejected
3) Trustee Decrypts $C_2$ and obtains SPID*, UID*, $X_{ab}$ and $n_2$
   $Trustee_{PR_{CSP}}(C_2) = (SPID*||UID*||X_{ab}||n_2)$
   If SPID*= =SPID&& UID*==UID and are valid?, then chooses a secrete number c (c<p) and computes K = $K_j=X_{ab}^{c}$ mod p,
   $X_c = g^c$ mod p,$X_{bc}=\hat{e}(X_b, X_c)$ mod p, andperformsC$_3$= $(e_K(ERN||n_3)||X_{ab}||X_{bc})$
   Trustee$\xrightarrow{C_3}U_i$
   If UID* or SPID* is not found with trustee, then theauthentication request will be rejected.
4) $U_i$ Computes $K=X_{bc}^{a}$mod p and obtains
   $d_K(C_3)=(ERN\ ||n_3)$
   if $n_3$==$n_1$+2, then inputs PWD* and BF*
   Decrypts $d_{BF*}$(ERN)=RN
   Finds $h_3$(PWD*+salt), $h_3$(RN⊕BF*)
   Performs $C_4=e_K(HPWD*||HBF*||n_4)$.
   $U_i \xrightarrow{C_4}$Trustee
   If $n_3 \neq n_1+2$, then the authentication process will be terminated.
5) Trustee Decrypts$C_4$ and obtains HPWD*, HBF* and $n_4$
   $d_K(C_4)= (HPWD*||HBF*||n_4)$
   If $n_4$== $n_3+1$, then computes $X_{ac}=\hat{e}(X_a, X_c)$ mod p$C_5=(e_K(HPWD*||HPWD||HBF*||HBF||X_a||n_5)||X_{ac})$ Trustee$\xrightarrow{C_5}CSP_j$
   if $n_4 \neq n_3+1$, then the authentication process will be terminated.
6) $CSP_j$ computes K= $X_{ac}^{b}$modp
   $d_K(C_5)=(HPWD*||HPWD||HBF*||HBF||X_{ac}||n_5)$ if((HPWD*==HPWD&&HBF*==HBF)&&($n_5$==$n_2$+3), then the cloud service will be provided to the user, otherwise authentication request will be rejected.

## 6. Trust and Reputation Management

There is no universally accepted definition for trust and reputation in the present ICT world, because it depends on particular jurisdiction and regulations. In the recent years, several trust and reputation management mechanisms have been proposed in. Each presents a partial view of trust and reputation and lacking in providing more accurate mathematical formal mechanisms. To measure

and predict the accurate trustworthiness of CSPs, we have proposed the chains of mathematical formal equations. By combining these equations we have calculated global trust value. In this process, an algorithm is proposed to select high trustworthy cloud service providers and to gains confidence in user control over the cloud computing management. We have presented a system-level trust and reputation evaluation model as depicted in Figure 2. In this model, a global trust value is computed from the genuine historical feedbacks received from the users about quality of cloud services, direct interactions, and direct and indirect reputation values.

1) **Service Trust of CSPs**

In the process of deriving mathematical formula for calculating global trust value, in this subsection, we first formulated the service trust in terms of historical feedbacks and direct interaction attributes.

i) Historical Feedbacks Trust (HFT): This is the amount of trust that one user has in CSP for a specific service usage and calculated based on the marsh mechanism[31, 32] via equation (6).

$$T(u, s)^t = I_0 * T_u(CSP)^{t-1} \quad (6)$$

Where, $I_0$ is the importance of feedbacks, $T_u(CSP)$ is the estimation of the historical feedback trust with respect to $T(u, s)^{t-1}$ in the past, with θ to t–1 time window,t indicates current time, s denotes service type and u represent end user. Only the feedback trust rating within that time window is taken for the aggregation. HFT is computed by equation (7).

$$T_u(csp)^{t-1} = (1/|A|)\Sigma_{A \epsilon s} T_u(csp, s) \quad (7)$$

Where, A is the number of feedbacks received for s within $\theta$ to t–1 temporal time window.

ii) Direct interaction attributes: We focused on the three kinds of trust attributes for calculating the direct interaction trust as follow:

a. Data Processing Trust is calculated based on whether the authentication process and user actions on service data are taken place successfully or not. Trustee stores the number of non-error requests ($R_{ne}$) and error number requests (Re). Data Processing Trust ($T_{dp}$) can be calculated through equation (8).

$$T_{dp} = (R_{ne} / (R_{ne+} R_e)) \quad (8)$$

b. Data Privacy Trust is computed based onwhether any unauthorized entities are permitted to access the confidential data or not and the governed privacy compliance laws are effective or not. Trustee stores the number $M_u$[1 or 0] that indicates misuse or unauthorized access and the rating given to the governed privacy compliance laws ($CL_r$) [0 to 9]. The Data Privacy Trust ($T_{priv}$) value can be computed using equation (9).

$$Pr = \alpha * Mu + (1 - \alpha) * CLr \quad (9)$$

$$T_{priv} = \begin{cases} 0 & \text{If } P_r = 0 \\ 1 & \text{If } P_r > 0 \end{cases}$$

c) Data Transmission Trust is calculated based on whether the communication data is transmitted successful or not. Trustee stores the number of successful data transmission requests ($DT_s$) and failure data transmission requests($DT_f$) and the data transmission trust calculated via equation (10).

$$T_{dt} = (DT_s / (DT_s + DT_f)) \quad (10)$$

Thus, the Service Trust value ($ST_{val}$) can be computed by combining the equations (6) to (10) and formed equation (11).

$$ST_{val} = T(u, s)^t + \Sigma (I_1 * T_{dp} + I_2 * T_{priv} + I_3 * T_{dt} \quad (11)$$

Where, $I_{i,\ 1 \le i \le 3}$ is the importance of various trust parameters and $\Sigma I_{i,\ 1 \le i \le 3} = 1$.

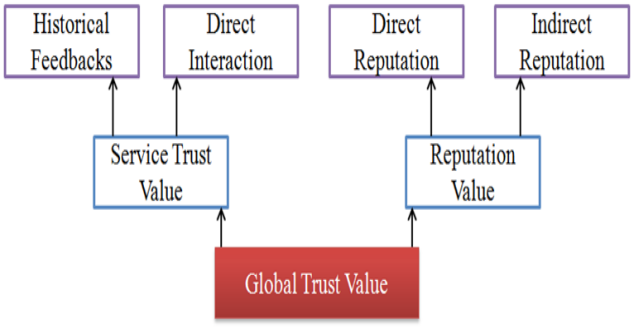


**Figure 2.** Trust and Reputation Evaluation Model.

**2) Service Reputation of CSPs**

In this subsection, we have formulated the mathematical equation for calculating service reputation of CSPs in terms of direct and indirect reputation values.

i) Direct Reputation: It is derived from the user direct interaction with cloud service providers. Direct reputation ($R_d$) can be calculated as the number of users who has chosen the service of the CSP (denoted as $N_u$)

divided by the number of users currently using the service (i.e., $N_u$'), where $N_u \geq N_u$'. The direct reputation value is computed by equation (12).

$$R_d = N_u' / N_u. \quad (12)$$

ii) Indirect Reputation: Indirect reputation is the reputation based on second-hand evidence. It deals with three kinds of reputation attributes, such as 1. Witness reputation 2. Neighbourhood reputation 3. Group derived reputation.

1. **Witness Reputation** is calculated using the evidence information gathered from the business partners and technology partners about service of CSP.
2. **Neighbourhood Reputation** is the reputation based on social prejudice
3. **Group derived reputation** is computed based on membership to certain group.

For each kind of indirect reputation rating we consider the range of numbers from 0 to 10 and the aggregation value of these three evidences is the indirect reputation.

Thus the reputation value is computed through equation (13).

$$\text{Reputation } (R_{val}) = \alpha * R_d + (1 - \alpha) * \text{Indirect Reputation} \quad (13)$$

Where, α is the importance of reputation values.

Therefore, the mathematical formal equation to measure the accurate global trustworthiness of the cloud service providers is formulated based on cloud service cost (SC) and equations (11) and (13). Thus, the global trust value is calculated as per equation (14).

$$\text{Global Trust value } (GT_{val}) = \alpha * (C_d / C_{daccrange}) + \beta * ST_{val} + \gamma * R_{val} \quad (14)$$

Where, α,β and γ are the user chosen weights for cost, trust and reputation values and $\alpha+\beta+\gamma=1$. Note: if $C_d$ is less than one and is greater than equal to negative $C_{daccrange}$ value, then $C_d$ is assigned with $C_{daccrange}$ or if $C_d$ is out of $C_{daccrange}$, then the value of $C_d/C_{daccrange}$ is considered as zero.

The selection process of appropriate and high trustworthy CSP is represented in the Algorithm 2: Cloud Service Provider Selection based on the Global Trust Value

### *Algorithm 2: Cloud Service Provider Selection based on the Global Trust Value*

**if** (user satisfied with some CSP's service attributes) then

User requests the CSP's trust and reputation values from the trustee.

**if** (($C_d \epsilon C_{daccrange}$) and (($St_{val} \geq$min acceptable service trust value) and ($R_{val} \geq$min acceptable reputation value)) then

User finds the global trust value ($Gt_{val}$) from equation (14) based on his/her importance to the service cost, trust and reputation.

**if**($Gt_{val}$ satisfies the user requirements and maximum) then

User interacts with the trustee for registration to get the cloud service from the desired CSP.

**repeat**

Legitimated user monitors the quality of services and records the evidences based on SLAs and PLAs attributes.

**until**(the service session is terminated).

Finally, user sends the service feedback to the trustee, then trustee finds the values of $T(u, s)^t$, $T_{dp}$, $T_{priv}$, $T_{dt}$, $R_d$ using equations (7), (8), (9), (10) and (12) and indirect reputation. From the equations (11) and (13), trustee calculates the cloud service provider trust and reputation values. Then trustee updates these values in the highly secured and distributed data bases.

## 7. Completeness of the Proposed Protocol

We used cryptographic GNY[33] belief logic to formally analyze the working nature of our trusted authentication mechanism and to verify whether our mechanism meets its goals.[33] GNY belief logic is the substantial extension of BAN logic.[34] First, we present the basic terminologies and statements, protocol transformation, goals and assumption list we used. Next, we describe the logical postulates adoption.

A. Basic Terminologies and Statements

Let $CP_i$ be the credential parameter message and the following basic terminologies are introduced on $CP_i$:

- $h(CP_i)$:hash operation on $CP_i$.
- $\{CP_i\}_{+K}$, $\{CP_i\}_{-K}$:$CP_i$ is encrypted with +K and decrypted with -K.
- $\{CP_i\}_K$, $\{CP_i\}^{-1}_K$:$CP_i$ is encrypted and decrypted with secrete key K.

**Statements:** Let $E_i$ and $E_j$ be two communication entities and the following statements are formed on $E_i$ and $E_j$.

- $E_i \lhd E_j$: $E_i$ holds $E_j$
- $E_i \ni CP_i$: $E_i$ possesses credential parameter message $CP_i$
- $E_i | \sim CP_i$: $E_i$ once conveyed $CP_i$

- $E_i|\equiv\#(CP_i)$: $E_i$ believes that $CP_i$ is fresh
- $E_i\equiv\phi(CP_i)$: $E_i$ believes that $CP_i$ is recognizable
- $E_i|\equiv E_i \overset{S}{\leftrightarrow} E_j$: $E_i$ believes that Sis a suitable secrete for $E_i$ and $E_j$
- $E_i|\equiv \overset{K}{\rightarrow} E_j$: $E_i$ believes that public key +K is suitable for $E_j$
- $E_i=>X$: $E_i$ has jurisdiction over X
- $E_i \lhd *X$: $E_i$ is told that he/she didn't conveyX previously in the current session.

B. Protocol Transformation

Our proposed multifactor authentication protocol is mapped into the form of $E_i \rightarrow E_j$:$CP_i$

1) $U_i \rightarrow CSP_j$:$\{\{UID^*||X_a||n_1\}_{+K}\}$
2) $CSP_j \rightarrow$Trustee:$\{\{SPID^*||UID^*||X_a||X_{ab}||n_2\}_{+K}\}$
3) Trustee $\rightarrow U_i$:$\{\{ERN\ ||n_3\}_K||X_{ab}||X_{bc}\}$
4) $U_i \rightarrow$Trustee:$\{\{HPWD^*||HBF^*||n_4\}_K\}$
5) Trustee$\rightarrow CSP_j$:$\{\{HPWD^*||HPWD||HBF^*||HBF||X_a||n_5\}_K||X_{ac}\}$

Parsing of the authentication protocol into $E_i|\sim CP_i$ and $E_i \lhd *X$ is given below.

1) $CSP_j \lhd *\{*UID, *X_a, *n_1\}+_K \sim > U_i|\equiv U_i \leftrightarrow CSP_j$
2) Trustee$\lhd *\{ *SPID*||*UID*||*X_a||*X_{ab}||*n_2\}_{+K} \sim > CSP_j| \equiv CSP_j \leftrightarrow$ Trustee
3) $U_i \lhd *\{\{_{+K}*ERN||*n_3\}_K||*X_{ab}||*X_{bc}\} \sim >$Trustee$|\equiv$ Trustee $\leftrightarrow U_i$
4) Trustee$\lhd *\{*HPWD*||*HBF*||*n_4\}_K \sim > U_i|\equiv_{+K} U_i \leftrightarrow$ Trustee
5) CSP$\lhd *\{HPWD*||*HPWD_{+K}||HBF*||*HBF||*X_{ac}||*n_5\}_K \sim >$Trustee$|\equiv$ Trustee $\leftrightarrow CSP_j$

C. Goals

The followings are the goals which describe the basic functionalities of the proposed protocol.

1) Authentication on message content

In the proposed protocol, $CSP_j$ believes that the user login request message contents are recognizable and valid

$$CSP_j|\equiv \phi\ \{\{UID^*||X_a||n_1\}_{+K}\}.$$

In the second flow, Trustee believes that $CSP_j$ message contents are recognizable and valid

$$\text{Trustee}\ |\equiv \phi\ \{\{SPID^*||UID^*||X_a||X_{ab}||n_2\}_{+K}\}.$$

In the third flow, $U_i$ believes that the Trustee message contents are recognizable and valid

$$U_i|\equiv \phi\ \{\{ERN\ ||n_3\}_K\ ||X_{ab}||X_{bc}\}.$$

In fourth flow, Trustee believes that $U_i$ reply message contents are recognizable and valid

$$\text{TTP}\ |\equiv \phi\{\{HPWD^*||HBF^*||n_4\}_K\}.$$

In fifth flow, $CSP_j$ believes that the Trustee reply message contents are recognizable and valid

$$CSP_j|\equiv \phi\{\{HPWD^*||\ HPWD||HBF^*||HBF||X_{ac}||n_5\}_K\}.$$

2) Authentication on message origin

From the login request, $CSP_j$ believes that $U_i$ is originated

$$CSP_j\ |\equiv U_i|\sim\{\{UID^*||X_a||n_1\}_{+K}\}.$$

In the second flow, Trustee believes $CSP_j$ is originated message

$$\text{Trustee}|\equiv CSP_j|\sim\{\{SPID^*||UID^*||X_a||X_{ab}||n_2\}_{+K}\}.$$

In the third flow, $U_i$ believes Trustee is replied

$$U_i|\equiv \text{Trustee}|\sim\{\{ERN\ ||n_3\}_K||X_{ab}||X_{bc}\}.$$

In the fourth flow, Trustee believes $U_i$is replied

$$\text{Trustee}|\equiv U_i|\sim\{\{HPWD^*||HBF^*||n_4\}_K\}.$$

In the fifth flow, $CSP_j$ believes and validates Trustee response

$$CSP_j|\equiv \text{Trustee}|\sim\{\{HPWD^*||HPWD||HBF^*||HBF||X_a||n_5\}_K||X_{ac}\}.$$

3) Mutual Identity Verification

From the first flow, $CSP_j$ verifies UID, if it matches then $CSP_j$ sends SPID* and UID* to Trustee, otherwise user request will be terminated

$$CSP_j|SU_i \ni (UID^*).$$

From the second flow, Trustee believes and verifies SPID* and UID*, if SPID* and UID* are found and valid, then Trustee sends the intermediate secretes and encrypted random number (ERN) to $U_i$, otherwise authentication request will be terminated

$$\text{Trustee}|\equiv U_i \ni (UID^*)\ \&\&CSP_j \ni (SPID^*).$$

From the third flow, $U_i$ verifies $CSP_j$ and Trustee incremented nonce $n_3$, if$n_3==n_1+2$, then user believes that the response received is genuine, otherwise authentication process will be stopped

$$U_i|\equiv \text{Trustee}, CSP_j \ni (n_3).$$

From the fourth flow, Trustee verifies $U_i$ incremented nonce $n_4$, if$n_4==n_3+1$, then Trustee believes that the response received from $U_i$ is genuine, otherwise the authentication process will be terminated

$$\text{Trustee}|U_i \ni (n_4).$$

From the fifth flow, $CSP_j$ verifies Trustee incremented nonce $n_5$, if $n_5==n_2+3$, then $CSP_j$ believes that the response received from Trustee is genuine and also verifies the user credential parameters; otherwise the authentication process will be terminated

$$CSP_j|\equiv U_i \ni (n_5, HPWD^*, HPWD, HBF^*, HBF).$$

D. Session Key Material Establishment

$U_i$, $CSP_j$ and Trustee believes each other that $X_b$, $X_b$ and $X_c$ are their intermediate secrete values for generating shared session key

$$U_i|\equiv Truestee\ |\equiv CSP_j\ |\equiv \{U_i, CSP_j, TTP\} \ni \{X_b, X_b, X_c\}.$$

$U_i$, $CSP_j$ and Trustee believes that K is a shared one-time secrete key for the current session

$$U_i|\equiv CSP_j|\equiv Trustee|\underset{K}{\equiv}\{U_i \leftrightarrow CSP_j,\ CSP_j \leftrightarrow TTP,\ U_i \leftrightarrow Trustee\}.$$

D. Assumption List

We consider the following assumptionsin our cryptographic authentication protocol.

- Trustee chooses a random values as private key–K, computes corresponding public key +K and prepares a one-time intermediate secrete value $X_c$ for generating shared session key

$$Trustee \ni -K,\ Trustee \ni +K,\ Trustee \ni X_c.$$

- Trustee publishes a public key +K for the users and CSPs to encrypt their communication messages and also believes that +K is suitable for $CSP_j$ and $U_i$.

$$Trustee\ |\equiv \xrightarrow{+K}\{CSP_j, U_i\}.$$

- $CSP_j$ chooses a random value as private key–K, computes corresponding public key +K and prepares a one-time intermediate secrete value $X_b$ for generating shared session key

$$CSP_j \ni -K,\ CSP_j \ni +K,\ CSP_j \ni X_b.$$

- $CSP_j$ publishes a public key +Kfor the users to encrypt their communication parameters and believes that +K is suitable for$U_i$.

$$CSP_j\ |\equiv \xrightarrow{+K}\{U_i\}.$$

- $U_i$ chooses a random value 'a' and prepares one-time intermediate secrete $X_a$. $U_i$ believes that $X_a$ is fresh and it will be used by Trustee and $CSP_j$ to compute shared secrete keys

$$U_i \ni X_a,\ U_i \equiv \#(X_a).$$

- $CSP_j$ chooses a random value 'b' and prepares one-time intermediate secrete $X_b$. $CSP_j$ believes that $X_b$ is fresh and it will be used byTrustee and $U_i$ to compute shared secrete keys

$$CSP_j X_b,\ CSP_j \equiv \#(X_b).$$

- Trustee chooses a random value 'c' and prepares one-time intermediate secrete value $X_c$. Trustee believes that $X_c$ is freshand it will be used by $CSP_j$ and $U_i$ to calculate shared secrete keys

$$Trustee X_c,\ Trustee \equiv \#(X_c).$$

F. The Logical Postulates

In this subsection, we describe our authentication protocol functionalities by using suitable GNY logic postulates as follows.

1) The first flow:

$$CSP_j \triangleleft \{UID^*||X_a||n_1\}_{+K},\ CSP_j \ni -K$$

$$CSP_j \triangleleft (UID^*, X_a, n_1),\ U_i \ni (UID^*, X_a, n_1)$$

If $U_i$ is told that the first message $\{UID^*||X_a||n_1\}$ is encryted with $CSP_j$ public key +K, then $CSP_j$ can obtains UID*, $X_a$ and $n_1$ using corresponding private key –K. $U_i$ possess UID*, $X_a$ and $n_1$ for further interactions in the current session.

$$CSP_j\ |\equiv \phi\ (UID^*, X_a, n_1),\ CSP_j \ni -K,$$

$$CSP_j\ |\equiv \#\ (UID^*, X_a, n_1)$$

$$CSP_j\ |\equiv \phi\ \{UID^*||X_a||n_1\}_{+K},\ CSP_j\ |\equiv \#\{UID^*||X_a||n_1\}_{+K}$$

If $CSP_j$ decrypts $U_i$ sent message using private key –K,then, $CSP_j$ believes thatUID*, $X_a$andn$_1$are recognizable and fresh. If UID*==UID, then $CSP_j$ believe that the values of $X_a$andn$_2$are never sent in previous sessions and initiates further process.

2) The second flow:

$$Trustee \triangleleft \{SPID^*||UID^*||X_a||X_{ab}||n_2\}_{+K},\ Trustee \ni -K$$

$$Trustee \triangleleft (SPID^*, UID^*, X_a, X_{ab}, n_2),$$
$$Trustee \ni (SPID^*, UID^*, X_a, X_{ab}, n_2)$$

If $CSP_j$is told that the second message $\{SPID^*||UID^*||X_a||X_{ab}||n_2\}$is encryted with Trusteepublic key +K, thenTrustee obtains SPID*, UID*, $X_a$, $X_{ab}$andn$_2$using private key –K. $CSP_j$possess SPID*, $X_b$andn$_2$ for further interactions in the current session.

$$\text{Trustee} \mid\equiv \phi\ (\text{SPID*}, \text{UID*}, X_a, X_{ab}, n_2)\ \text{Trustee} \ni -K,$$
$$\text{Trustee} \mid\equiv \#\ (\text{SPID*}, \text{UID*}, X_a, X_{ab}, n_2)$$

$$\text{Trustee} \mid\equiv \phi\ \{\text{SPID*}||\text{UID*}||X_a||X_{ab}||n_2\}_{+K},$$
$$\text{Trustee} \mid\equiv \#\ \{\text{SPID*}||\text{UID*}||X_a||X_{ab}||n_2\}_{+K}$$

If Trustee decrypts $CSP_j$ sent message using private key –K, then Trustee believes that SPID*, UID*, $X_a$, $X_{ab}$ and $n_2$ are recognizable and fresh. If SPID* and UID*are found and valid, then Trustee is entitled to believe that the values of $X_a$, $X_{ab}$ and $n_2$ are never sent in previous sessions. Trustee believes $U_i$ and $CSP_j$ and continues further authentication process.

3) The third flow:

$$U_i \triangleleft \{\{\text{ERN}\ ||n_3\}_K||X_{ab}||X_{bc}\},\ U_i \ni K$$

$$U_i \triangleleft (\text{ERN}, n_3, X_{abc}),\ U_i \ni (\text{ERN}, n_3, X_{abc})$$

If Trustee is told that the third message components $\{\text{ERN}\ ||n_3\}_K$ are encryted with the shared secrete key K, then$U_i$ finds K using intermediate secretes and obtains ERN and $n_3$using K.

$$U_i \triangleleft^* \{\{\text{ERN}\ ||n_3\}_K||<X_{ab}>||X_{bc}\},\ U_i \mid\equiv U_i \leftrightarrow \text{Trustee},$$
$$U_i \mid\equiv \phi(\text{ERN}, n_3), U_i \mid\equiv \#(\text{ERN}, n_3, K)$$

$$U_i \mid\equiv \text{Trustee} \ni \{\text{ERN}\ ||n_3\}_K,\ U_i \mid\equiv \text{Trustee} \ni K$$

Below given conditions are hold: (1) $U_i$ receives a message$\{\{\text{ERN}\ ||n_3\}_K||<X_{ab}>||X_{bc}\}$which consists of $\{\text{ERN}\ ||n_3\}_K$ and intermediate secretes $X_{ab}$ and $X_{bc}$; (2) $U_i$ calculates current session key K using intermediate secrets;(3) $U_i$ believes that Trustee has K; (4) $U_i$ believes all the decrypted and intermediate secret components are recognizable;(5) $U_i$ believesthat Trustee sent message and K are fresh; (6)$U_i$ checks $n_3$ with $n_1$+2, if it is equal, then $U_i$ believes that the Trustee is legitimate entity. Thus $U_i$ is entitled to believe that Trustee once conveyed a message $\{\{\text{ERN}\ ||n_3\}_K||<X_{ab}>||X_{bc}\}$; Trustee once conveyed message components ERN and $n_3$ are encrypted with K; and Trustee possesses K.

4) The fourth flow:

$$\text{Trustee} \triangleleft \{\text{HPWD*}||\text{HBF*}||n_4\}_K,\ \text{Trustee} \ni K$$

$$\text{Trustee} \triangleleft \{\text{HPWD*}, \text{HBF*}, n_4\},$$
$$\text{Trustee} \ni \{\text{HPWD*}, \text{HBF*}, n_4\}$$

If $U_i$ is told that the fourth message $\{\text{HPWD*}||\text{HBF*}||n_4\}_K$ is encryted with a shared secret key K, then Trustee obtains HPWD*, HBF* and $n_4$ using K.

$$\text{Trustee} \triangleleft^* \{\text{HPWD*}||\text{HBF*}||n_4\}_K,\ \text{Trustee} \mid\equiv \text{Trustee} \leftrightarrow U_i,\ \text{Trustee} \mid\equiv \phi(\text{HPWD*}, \text{HBF*}, n_4),$$
$$\text{Trustee} \mid\equiv \#(\text{HPWD*}, \text{HBF*}, n_4, K)$$

$$\text{Trustee} \mid\equiv U_i \ni \{\text{HPWD*}||\text{HBF*}||n_4\}_K,\ \text{Trustee} \mid\equiv U_i \ni K$$

Below given conditions are hold: (1) Trustee receives a message$\{\text{HPWD*}||\text{HBF*}||n_4\}_K$ that is encrypted with K; (2) Trusteebelieves that the secrete key K is suitable for $U_i$; (3)Trustee believes all the decrypted components are recognizable;(4)Trustee believes that $U_i$ sent message and K are fresh; (5)Trustee checks $n_4$ with $n_3$+1, if it is equal, then Trustee believes that $U_i$ is legitimate entity. Then Trustee is entitled to believe that $U_i$ once conveyed a message $\{\text{HPWD*}||\text{HBF*}||n_4\}_K$; $U_i$ once conveyed message $\{\text{HPWD*}||\text{HBF*}||n_4\}_K$ is encrypted with K; and $U_i$ possesses K.

5) The fifth flow:

$$CSP_j \triangleleft \{\text{HPWD*}||\ \text{HPWD}||\text{HBF*}||\text{HBF}||X_{ac}||n_5\}_K,$$
$$CSP_j \ni K$$

$$CSP_j \triangleleft \{\text{HPWD*}, \text{HPWD}, \text{HBF*}, \text{HBF}, n_4\},$$
$$CSP_j \ni \{\ \text{HPWD*}, \text{HPWD}, \text{HBF*}, \text{HBF}, n_4\}$$

If Trusteeis told that the fifth message $\{\text{HPWD*}||$ $\text{HPWD}||\text{HBF*}||\text{HBF}||X_{ac}||n_5\}_K$ is encryted with the shared secret key K, then $CSP_j$ obtains HPWD*, HPWD, HBF*, HBF and $n_4$ using K.

$$CSP_j \triangleleft^* \{\text{HPWD*}||\ \text{HPWD}||\text{HBF*}||\text{HBF}||X_{ac}||n_5\}_K,\ CSP_j$$
$$\mid\equiv CSP_j \leftrightarrow \text{Trustee},\ CSP_j \mid\equiv \phi(\text{HPWD*}, \text{HPWD}, \text{HBF*},$$
$$\text{HBF}, X_{ac,}\ n_5),\ CSP_j \mid\equiv \#(\text{HPWD*}, \text{HPWD}, \text{HBF*}, \text{HBF},$$
$$X_{ac,}\ n_5, K)$$

$$CSP_j \mid\equiv \text{Trustee} \ni \{\text{HPWD*}||\ \text{HPWD}||\text{HBF*}||\text{HBF}||X_{ac}||n_5\}_K,\ CSP_j \mid\equiv \text{Trustee} \ni K$$

Below given conditions are hold: (1) $CSP_j$ receives a message$\{\text{HPWD*}||\ \text{HPWD}||\text{HBF*}||\text{HBF}||X_{ac}||n_5\}_K$ that is encrypted with K; (2) $CSP_j$ believes that K is a current session key for himself and Trustee; (3) $CSP_j$ believes that decrypted parameters are recognizable;(4) $CSP_j$ believes that Trustee sent message and K are fresh; (5)$CSP_j$ checks $n_5$with $n_2$+3, if it is equal, then $CSP_j$believes that the Trustee is legitimate entity. Then $CSP_j$ is entitled to believe that Trustee once conveyed a message $\{\text{HPWD*}||\text{HPWD}||\text{HBF*}||\text{HBF}||X_{ac}||\ n_5\}_K$; Trustee conveyed message is encrypted with K; and Trustee possesses K.

## 8. Performance Evaluation

This section reports comparative study, computational efficiency, selection of trustworthy CSPs and fingerprint samples recognition rates. In first subsection, we presented the comparative study with the existing schemes. Next, we analysed the computational efficiency of our scheme with the existing base scheme. By using two case studies we analysed the selection of desired and trustworthy cloud service provider in the second subsection. Various benchmark database fingerprint samples recognition rates are reported in the third subsection.

**Setup:** We have implemented our proposed investigation on a computer which has windows 7 operating system with 4GB RAM and 2.0GHz Intel Core i7 processor. C#.NET framework was installed on this computer which contains Visual Studio community 2013 as a frontend, SQL Server 2012 R2 SP1 as a backend and a Windows Azure Emulator as software platform.

### 8.1 Comparisons and Computational Efficiency

In this subsection, first we compare our trusted authentication protocol with the existing mechanisms in terms of mutual authentication, resistant to various attacks, trust and reputations management as given in Table 2. The existing authentication mechanisms are effortless to enable the trust and reputation management with user registration. Existing trust evaluation approaches in are lacking in designing and developing mathematically formal framework for trust and reputation management. The mechanism proposed in is suitable for mutual authentication and supports reply attacks. The biometric fingerprint-based scheme proposed in supports reply and clock synchronization attacks.

**Table 2.** Comparision with existing mechanisms

| | $CP_1$ | $CP_2$ | $CP_3$ | $CP_4$ | $CP_5$ | $CP_6$ | $CP_7$ | $CP_8$ | $CP_9$ |
|---|---|---|---|---|---|---|---|---|---|
| A.J.Choudhury et al.[14] | Yes | No | Yes | No | No | No | No | No | No |
| J.Yanget al.[15] | No | No | Yes | Yes | No | No | No | No | No |
| Hong Liu et al.[19] | No | No | Yes | Yes | Yes | No | No | No | No |
| J. Zhouet al.[20] | Yes | No | Yes | Yes | Yes | No | No | No | No |
| Jia-Lun Tsai et al.[21] | No | No | Yes | No | Yes | No | No | No | No |
| Xiaoyong li et al.[28] | No | No | No | No | Yes | No | No | No | No |
| ABarsoum et al.[30] | No | No | No | No | Yes | No | No | No | No |
| J. Huang et al.[29] | No | No | No | No | Yes | No | No | No | No |
| Our's | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |

$CP_1$: Mutual authentication
$CP_2$: Mathematically formal trust and reputation
$CP_3$: Resistance to replay attack
$CP_4$: Resistance to impersonation attack
$CP_5$: Suitability to multiple service providers
$CP_6$: White-washing attack
$CP_7$: Collusion attack
$CP_8$: Bad mouthing attack
$CP_9$: Good mouthing attack

The mechanisms described in are not suitable for collaborative cloud service providers. Existing mechanisms proposed in are suitable for multiple service providers and are immune to the reply and clock synchronization attacks. The schemes also support mutual authentication. The mechanisms presented in do not support trust and reputation management as well as vulnerable to white-washing, collusion, bad mouthing and good mouthing attacks. In, the proposed mechanisms do not support mutual authentication. However, the mechanisms presented in are vulnerable to the reply, good mouthing, bad mouthing, collusion and white-washing attacks. Therefore, our investigation meets all the design goals and is immune to various reply and impersonation attacks.

Next, we compare the computation costs of our authentication protocol with the existing base scheme. Let $T_{bp}$ be the bilinear Diffie-Hellman pairing operation time, $T_h$ is one-way hash operation time, $T_c$ and $T_x$ are concatenation and

Exclusive-OR operation times and, $T_d$ and $T_m$ are division and multiplication operation times respectively. The comparison of computation cost of our proposed authentication protocol with an existing mechanism is given in Table 3.

**Table 3.** Computation cost comparision with existing scheme

| Phase | Party | Existing Scheme Time Consumption | Our scheme Time Consumption |
|---|---|---|---|
| **Registration** | User | $T_{bp}+T_h+T_m+T_d+T_c+T_x$ | $2T_h+T_c+T_x$ |
| | CSP | $T_{bp}+T_h+T_m+T_d+T_c+T_x$ | $2T_h+T_c+T_x$ |
| **Authentication** | User | $4T_{bp}+5T_h+3T_m+T_d+3T_c+T_x$ | $2T_{bp}+2T_h+6T_c+T_x$ |
| | Trustee | Nil | $3T_{bp}+2T_c$ |
| | CSP | $4T_{bp}+4T_h+T_m+2T_c+T_x$ | $3T_{bp}+9T_c$ |
| **Time Complexity** | | **10Tbp+11Th+ 6Tm +3Td + 7Tc+4Tx** | **8Tbp+6Th+ 19Tc+3Tx** |

In general, concatenation and bitwise Exclusive-OR operations are much faster and will consume constant timing, so that these two operations time can be neglected in calculating computation cost. Therefore, for registration process, our scheme requires only four hash operations (i.e., $4T_h$). On the other hand scheme consumes two bilinear Diffie-Hellman pairing, two hash, two multiplication and two division operations (i.e., $2T_{bp}+2T_h+2T_m+2T_d$). For authentication process, our authentication protocol consumes $8T_{bp}+2T_h$ and scheme requires $8T_{bp}+9T_h+4T_m+T_d$. So, the total computation cost of our authentication mechanism is $O(8T_{bp}+6T_h)$and scheme consumes $O(10T_{bp}+11T_h+6T_m+3T_d)$. Therefore, we can conclude that our proposed authentication scheme is computationally efficient and robust towards various reply and impersonation attacks than the existing schemes.

## 9. Trustworthy CSPs Selection

In this subsection, we describe the selection of desired and trustworthy cloud service providers using two case studies. In these two case studies, cloud users ($U_i$'s) can find trustworthy cloud service providers ($CSP_i$'s) by calculating global trust value from equation (14) and by following Algorithm 2. In equation (14), the values of service trust and reputation of $CSP_j$ can be computed from equation (11) and equation (13).

Case Study 1: Table 4 represents three cloud service users and their qualified cloud service providers based on the service cost difference range, minimum acceptable service trust value and minimum acceptable service reputation values. Table 4 also illustrates the different cloud service providers cost, service trust and reputation values. By using Table 5 and an algorithm 2, a user $U_i$ can find the desired trustworthy service provider $CSP_j$ based on the chosen weights as given in Weight_set_1 represents three users and their choice of trustworthy CSPs as given in Table 5, where user $U_1$selects$CSP_3$by assigning weight 1/3 to service cost, trust and reputation values. Similarly, $U_2$ and $U_3$ can select $CSP_2$ and $CSP_1$ respectively by assigning their desired weights. Weight_set_2 represents three users and their chosen trustworthy CSPs as given in Table 6 where user $U_1$can select $CSP_1$ because $CSP_1$ has more trust value than $CSP_2$ and $CSP_3$. Similarly, $U_2$ and $U_3$ can select $CSP_2$and $CSP_3$ respectively, based on the importance of cloud service reputation and cost values.

Case Study 2: In this case study, the parameters $C_d$, $ST_{val}$, $R_{val}$, $C_{drange}$, $ST_{minaccval}$, $R_{minaccval}$,α, β, and γare assigned with random values for the selection of cloud service providers. We first considered three hundred genuine cloud users and one thousand cloud service providers. Each User ($U_i$) filters CSPs based on their service attributes like {ST∋DT, SS≥DS, PS≥RS, SC≤SP, AT⊇RAT} and using algorithm 2, where three hundred random weight sets are used for specifying the importance of cloud service cost, trust and reputation. After filtration process, three hundred qualified CSPs satisfy the users attribute requirement. Three hundred random weight sets for service cost, trust and reputation and corresponding choice of CSPs are illustrated in Figure 3. Next, we considered one thousand genuine cloud users and one thousand corresponding choice of CSPswho satisfy only the users cloud service cost requirement with thousand random weight sets are depicted in Figure 4. Similarly, we considered one thousand genuine users and one thousand corresponding choice of CSPs who satisfy only the users cloud service trust as well as only reputation requirements with thousand random weight sets are presented in Figure 5, 6 respectively.

**Table 4.** Parameters of Qualified CSP's

| | $C_d$ | $ST_{val}$ | $R_{val}$ | $C_{drange}$ | $ST_{minaccval}$ | $R_{minaccval}$ |
|---|---|---|---|---|---|---|
| $U_1 \leftrightarrow CSP_1$ | 8 | 0.9 | 0.8 | [−10, 10] | 0.8 | 0.6 |
| $U_1 \leftrightarrow CSP_2$ | 9 | 0.8 | 0.7 | [−10, 10] | 0.8 | 0.6 |
| $U_1 \leftrightarrow CSP_3$ | 9 | 0.8 | 0.9 | [−10, 10] | 0.8 | 0.6 |
| $U_2 \leftrightarrow CSP_1$ | 9 | 0.8 | 0.8 | [−10, 10] | 0.8 | 0.6 |
| $U_2 \leftrightarrow CSP_2$ | 9 | 0.9 | 0.8 | [−10, 10] | 0.8 | 0.6 |
| $U_2 \leftrightarrow CSP_3$ | 8 | 0.8 | 0.9 | [−10, 10] | 0.8 | 0.6 |
| $U_3 \leftrightarrow CSP_1$ | 9 | 0.8 | 0.8 | [−10, 10] | 0.8 | 0.6 |
| $U_3 \leftrightarrow CSP_2$ | −7 | 0.8 | 0.7 | [−10, 10] | 0.8 | 0.6 |
| $U_3 \leftrightarrow CSP_3$ | 6 | 0.9 | 0.7 | [−10, 10] | 0.8 | 0.6 |

**Table 5.** Weight_set_1

| | $\alpha$ | $\beta$ | ã | Choice |
|---|---|---|---|---|
| $U_1$ | 1/3 | 1/3 | 1/3 | $CSP_3$ |
| $U_2$ | 1/2 | 1/4 | 1/4 | $CSP_2$ |
| $U_3$ | 1/5 | 2/5 | 2/5 | $CSP_1$ |

**Table 6.** Weight_set_2

| | $\alpha$ | $\beta$ | ã | Choice |
|---|---|---|---|---|
| $U_1$ | 0 | 1 | 0 | $CSP_1$ |
| $U_2$ | 0 | 0 | 1 | $CSP_3$ |
| $U_3$ | 1 | 0 | 0 | $CSP_2$ |

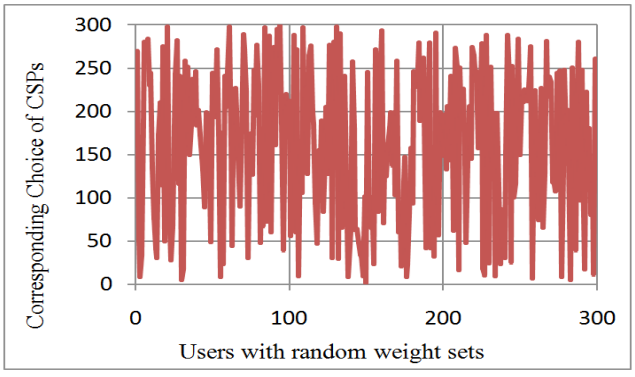


**Figure 3.** Different users and their corresponding choice of cloud service providers with the required satisfaction of cost, trust and reputation.

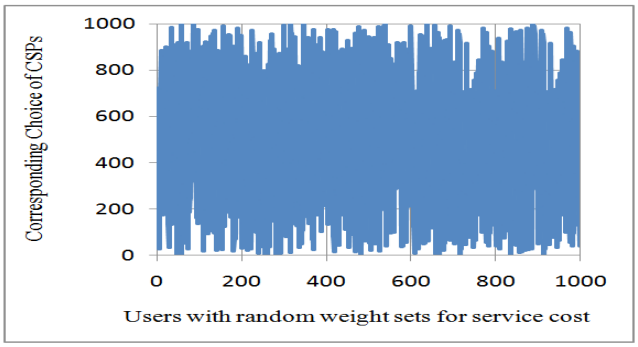


**Figure 4.** Different users and their corresponding choice of cloud service providers with the required satisfaction of only service cost..

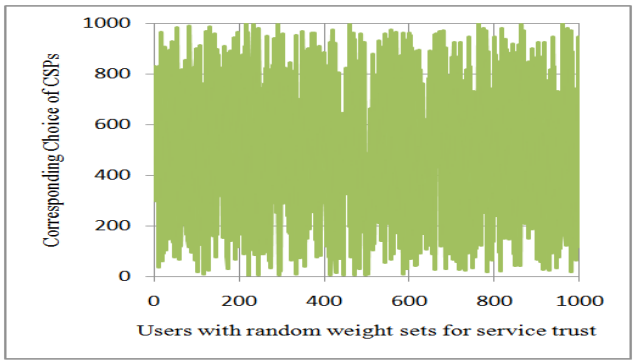


**Figure 5.** Different users and their corresponding choice of cloud service providers with the required satisfaction of only service trust.

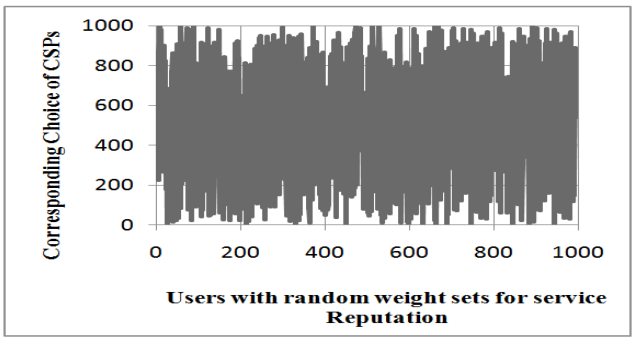


**Figure 6.** Different users and their corresponding choice of cloud service providers with the required satisfaction of only service reputation.

## 10 Fingerprint Samples Recognition Rates

### 10.1 Databases

To measure the robustness of our fingerprint-based authentication protocol, we have taken over 28,000 real and fake fingerprint images from four benchmark

databases available on the web.[35-39] In our investigation to evaluate the robustness, first we used FVC 2006 database. Fingerprint samples were collected from 150 heterogeneous participants including elderly people, academic and industrialists. FVC2006 database contained1800 images of 150 fingers, in-depth 12 samples.

Next we used CASIA-FingerprintV5 database and it has 20,000 fingerprint samples of 500 subjects, where each participant contributes 40 samples of eight fingers of both the hands except little fingers and five samples per finger. Fingerprint samples were collected from waiters, workers, graduate students, etc. Samples were of exaggerated rotations with various levels of pressure. The NIST database contains 2000 fingerprint image pairs with five classifications, such as whirl, tented arch, right loop, left loop and arch, where each pair of images is of completely different rolling. The ATVS-Fake Finger print Database contained two datasets comprises of more than 3,000 samples of 17 users. Samples were captured from both the hands of middle and index fingers.

We have set four time-window bounds such as 5, 10, 16 and 20 minutes on each of these databases to test the recognition correctness of fingerprint samples in terms of FNR and FPR. The recognition performance of our proposed authentication protocol for each of these four databases is reported in Figure 7, 8, where x-axis indicates benchmark databases and y-axis denotes percentages of FNR and FPR respectively. Figure 9 and 10 reports the false negative and false positive rates comparative study of our scheme with bio-metric fingerprint-based schemes. Our approach substantially produce better fingerprint recognition rate than the existing schemes.

We have found out the Average Equal Error Rates (AEERs) of each benchmark database and made comparative analysis with existing schemes as illustrated in

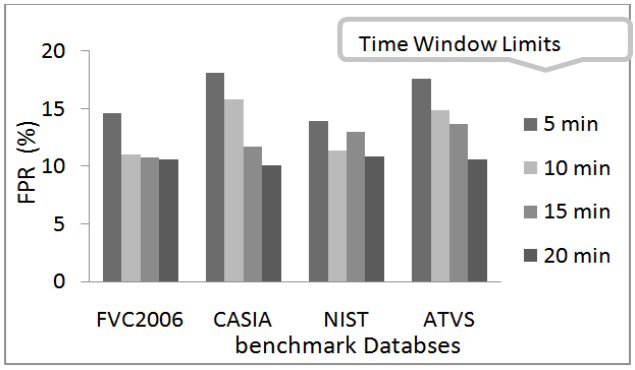


**Figure 7.** Performance of our authentication protocol in terms of False Negative Rate.

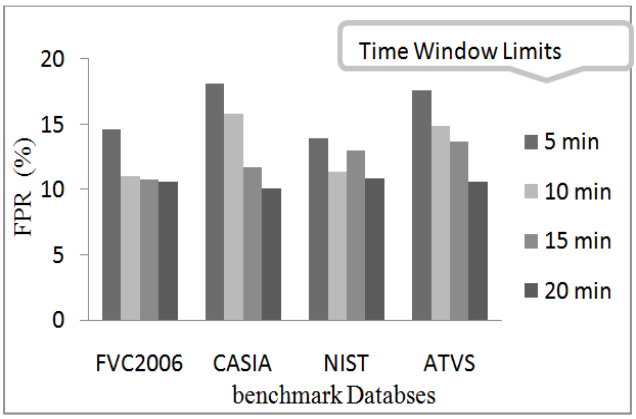


**Figure 8.** Performance of our authentication protocol in terms of False Positive Rate.

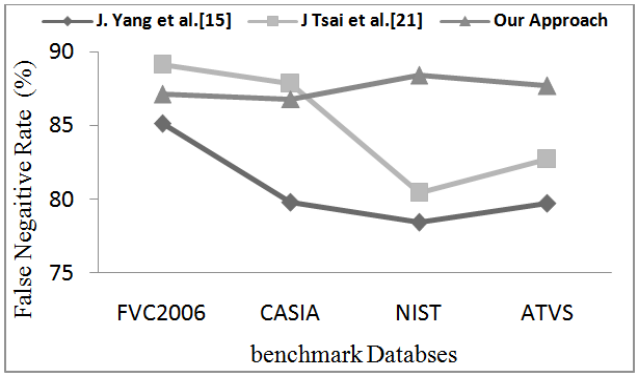


**Figure 9.** Performance comparison of our approach with existing fingerprint-based schemes in terms of False Negative Rate.

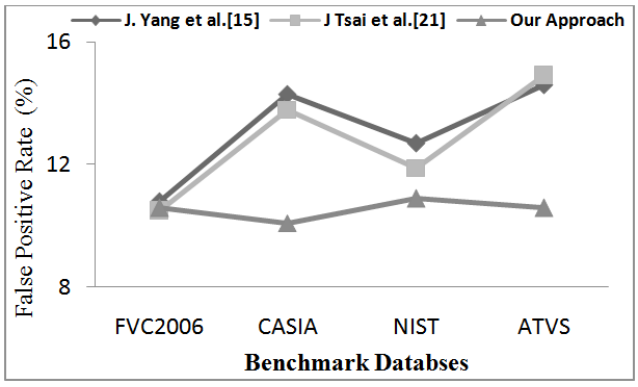


**Figure 10.** Performance comparison of our approach with existing fingerprint-based schemes in terms of False Positive Rate.

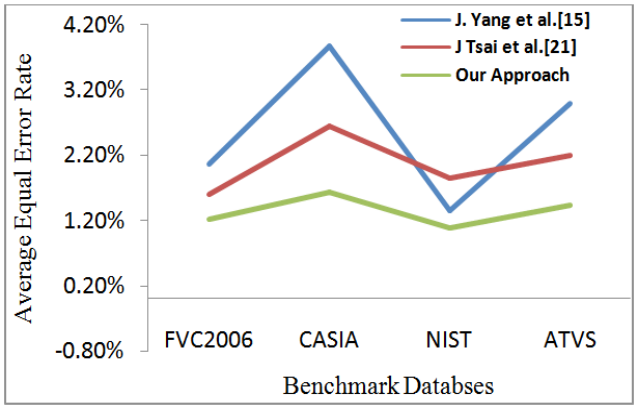


**Figure 11.** Performance analysis in terms of Equal Error Rates.

Figure 11. When AEER is taken as a fingerprint recognition evaluation parameter, it indicates the point where FNR and FPR coincide. An average equal error rate of proposed protocol is 1.34%. Our scheme has little variation in AEER of each input benchmark database than the existing schemes. As per the comparative analysis reported in Figure 11, we conclude that our approach produce better fingerprint samples recognition rates than the existing schemes.

## 11. Conclusion

In this article an authentication protocol is described for verifying mutual identities of cloud communication entities. The authentication is performed based on n-party diffie-Hellman bilinear pairing key distribution and random nonce. In this protocol, a cloud service provider verifies the user identities on hashed values. In addition, the details of the password and fingerprint data are never exposed to the service providers. The experimental analysis indicates that the proposed authentication protocol outperforms in computational efficiency and high-security strengths than the existing schemes. Next, we have presented mathematical formal mechanisms for calculating and managing trust and reputation values. In addition to formulating more accurate trust and reputation management mechanisms, we demonstrate the selection of appropriate and trustworthy cloud service providers using two case studies. This investigation can be further extended to relate to the amount of expected evidence and the amount of evidence collected in public distributed cloud computing.

## 12. Acknowledgements

Thanks to the Pondicherry University administration for providing required hardware and software resources to carry out this work successfully.

## 13. References

1. Smart Cities – Why, What, How, How? Available from: https://timstonor.wordpress.com/2013/06/06/smart-cities-why-what-how-how/. Date accessed: 06/06/2013.
2. An Expert's Guide to Oracle Technology. Available from: http://it.toolbox.com/blogs/oracle-guide/cloud-computing-defined-28433. Date accessed. 06/08/2013.
3. Mobile Cloud Applications. Available from: http://www.telecomasia.net/content/abi-cloud-computing-market-worth-. Date accessed: 06/08/2017.
4. Security and Privacy in Cloud Computing. Available from: www.cs.jhu.edu/~ragib/sp10/cs412.
5. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance. Available from: http://www.di.fc.ul.pt/~nuno/PAPERS/security3.pdf. Date accessed: 09/2009.
6. Amazon downplays report highlighting vulnerabilities in its cloud service. Available from: http://www.computerworld.com/s/article/9140074/Amazon_downplays_report_highlighting_vulnerabilities_in_its_cloud_service. Date accessed: 28/10/2009.
7. Qiu XF, Liu JW, Zhao PC. Secure cloud computing architecture on mobile Internet. International Conference on AIMSEC Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC); 2011. p. 619–22.
8. Itani W, Kayssi A, Chehab A. Privacy as a service: Privacy-aware data storage and processing in cloud computing architectures. Conference on Dependable Autonomic and Secure Computing International; 2009. p. 711–6.
9. Pearson S. Taking account of privacy when designing cloud computing services. Software Engineering Challenges of Cloud Computing; 2009. p. 44–52. Crossref.
10. Takabi JBD, Joshi G. Security and privacy challenges in cloud computing environments. IEEE Security Privacy. 2010; 8(6):24–31. Crossref.
11. Xiao Z, Xiao Y. Security and privacy in cloud computing. Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA). 2014; 15(2):843–59.
12. Ghazizadeh E. Trusted Computing Strengthens Cloud Authentication. The Scientific World Journal. 2014:1–18. Crossref.
13. Ullah S, Xuefeng Z. A Trusted Storage Architecture for Cloud Computing. International Journal of Advanced Science and Technology. 2014; 63(2014):65–72. Crossref.

14. Choudhury JA, Kumar P, Sain M, Lim H, Hoon JL. A Strong User Authentication Framework for Cloud Computing. IEEE Asia -Pacific Services Computing Conference; 2011. p. 110–5. Crossref.
15. Yang J. A fingerprint recognition scheme based on assembling invariant moments for cloud computing communications. IEEE Systems Journal. 2011; 5(4):574–83. Crossref.
16. Ruj S, Stojmenovic M, Nayak A. Decentralized access control with anonymous authentication of data stored in clouds. Transactions on Parallel and Distributed Systems. 2014; 25(2):384–94. Crossref.
17. Gong NZ, Wang D. On the Security of Trustee-Based Social Authentications. IEEE Transactions on Information Forensics and Security. 2014 Aug; 9(8):384–94. Crossref.
18. Chen CH, Yang TT, Chiang ML, Shih TF. A Privacy Authentication Scheme Based on Cloud for Medical Environment. Journal of Medical Systems. 2014; 143:1–16. Crossref.
19. Liu H, Ning H, Laurence QX, Yang T. Shared Authority Based Privacy-Preserving Authentication Protocol in Cloud Computing. IEEE Transactions on Parallel and Distributed Systems. 2015; 26(1):241–51. Crossref.
20. Zhou J, Lin X, Dong X, Cao Z, PSMPA: Patient Self-Controllable and Multi-Level Privacy-Preserving Cooperative Authentication in Distributed m-Healthcare Cloud Computing System. IEEE Transactions on Parallel and Distributed Systems. 2015; 26(6):1693–703. Crossref.
21. Tsai JL, Lo NW. A Privacy-Aware Authentication Scheme for Distributed Mobile Cloud Computing Services. IEEE Systems Journal. 2015; 9(3):805–15. Crossref.
22. Nagaraju S, Parthiban L. Provably Secure Multi-Factor Authentication for the Cloud Computing Systems. Indian journal of Science and Technology. 2016; 9(9):1–18. Crossref.
23. Nagaraju S, Parthiban L. Trusted framework for online banking in public cloud using multi-factor authentication and privacy protection gateway. Journal of Cloud Computing: Advances, Systems and Applications. 2015; 22:1–23.
24. Hosseini SS, Mohammadi D. Review Banking on Biometric in the World's Bank and Introducing a Biometric Model for Iran's Banking System. Proceedings of the Journal of Basic and Applied Scientific Research. 2012; 2(9):9152–60.
25. Ko RKL, pramanaet PJ. TrustCloud: A framework for accountability and trust in cloud computing. In Proceedings IEEE World Congress Services; 2011 Jul. p. 584–8.
26. Canedo ED. Trust Model for Reliable File Exchange in Cloud Computing. International Journal of Computer Science & Information Technology (IJCSIT). 2012; 4(1):1–18. Crossref.
27. Kuehnhausen M, Frost VS, Minden GJ. Framework for assessing the trustworthiness of cloud resources. In Proceedings IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support; 2012 Mar. p. 142–5. Crossref.
28. Li X, Junping DU. Adaptive and attribute-based trust model for service level agreement guarantee in cloud computing. IET Institution of Engineering and Technology Information Security. 2013; 7(1):39–50. Crossref.
29. Huang J, Nicol DM. Trust mechanisms for cloud computing. Journal of Cloud Computing: Advances systems and Applications. 2013, 2 (9), pp. 1-14.
30. Barsoum, Hasan A. Enabling dynamic data and indirect mutual trust for cloud computing storage systems. IEEE Transactions on Parallel & Distributed Systems. 2013; 24(12):2375–85. Crossref.
31. Ries S, Kangasharju J, Muhlhauser M. A Classification of Trust Systems. OTM One Time Mandate Workshops; 2006. p. 894–903. Crossref.
32. Burrows MA, Needham R. Logic of authentication. ACM Association for Computing Machinery Transaction Computer System. 1989; 23(5):1–13. Crossref 1, 2, 3.
33. Nessett M. A critique of the Burrows, Abadi, and Needham logic. Operating System Review. 1990; 24(2):35–8. Crossref.
34. Gong L, Needham R, Yahalom R. Reasoning about belief in cryptographic protocols. In Proceeding IEEE Computer Society Research in Security and Privacy; 1990. p. 234–46. Crossref.
35. A List of Fingerprint Databases Available on the Web. Available from: http://www.advancedsourcecode.com/fingerprintdatabase.asp. Date accessed: 12/04/2012.
36. Cappelli R, Ferrarab M, Franco A, Maltoni D. Fingerprint verification competition. Biometric Technology Today. 2006; 15(7-8):7–9. Crossref.
37. CASIA-FingerprintV5. Available from: http://biometrics.idealtest.org/. Date accessed: 03/10/2017.
38. NIST Fingerprint Database. Available from: https://www.nist.gov/srd/nist-special-database-4. Date accessed: 27/08/2010.
39. ATVS-FakeFingerprint Database. Available from: http://biometrics.idealtest.org/dbDetailForUser.do?id=11. Date accessed: 01/08/2017.