Data Security In Cloud via A Novel Cryptography Based Privacy Enhancing Scheme

S. Jayaprakash, Assistant Professor in Computer science, Sir Issac Newton College of Arts and Science, Nagapattinam, Tamilnadu, India, jayapraksh_nagai@yahoo.co.in

C. S. Rajarajeswari, Assistant Professor in Computer science, Indira Gandhi College of Arts and Science, Kadirkamam, Puducherry, India, govi07@yahoo.com

V.M.Suresh, Assistant Professor in Information Technology, E.G.S.Pillay Engineering College Nagapattinam, Tamilnadu, India, vmsureshme@gmail.com

Abstract - The extensive espousal of cloud services has raise wide range of security and privacy concerns that IT industries must take into account while they automate and standardize their service delivery. In this article, we propose a new crytopgraphybased privacy enhancing security solution (PESS) for cloud computing environment. PESS is implemented using a non-bilinear group signature method to deliver the anonymous authentication to pooled resources and cloud services. This solution provides an anonymous access for registered customers. Therefore, customers personal and sensitive information (e.g. age, gender, contact information, valid registration, payroll, benefits data and physical location etc.) can be verified without exposing their identity and customers can utilize cloud services with out describing their activities. Nevertheless, if a customer violates the rules framed by the cloud provider their access right is canceled. PESS offers anonymity, confidentiality, integrity and unlinkability of transferred information. Further more, we study other existing privacy-enhancing methods to access cloud offerings. This research relates the efficiency of PESS with other existing cryptographic methods. Overall, the study results indicate that PESS is more efficiency and provides better privacy as compared to the other three solutions given in the literature.

Keywords — Authentication, cloud computing, cryptography, data privacy, group signatures, security.

I. INTRODUCTION

Cloud computing now is everywhere cloud services have been envisioned as important elements of current information and networking technologies and step into our day-to-day lives. As stated in NIST, cloud computing is described as "A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [1]. In practice, a majority of cloud service applications involves high-performance computing to fulfill the needs of the customers. Some cloud services like open nebula, Cloud Safe net, Box.net, and Amazon Web Services (AWS) use customer identity, personal information and the physical location to provide cloud services. Hence, the cloud service providers have to experience various issues mainly associated with security and privacy of customers. Customers, who store their sensitive personal information (SPI), have a fundamental right to privacy. In the cloud computing environment, SPI includes information from different disciplines like financial data, health reports etc.

Security and privacy are the vital concerns in cloud computing which have less research focus. There are only limited cryptographic methods (e.g. anonymous access, GSM, zero-knowledge protocols etc.) available to offer anonymous access. The cloud service providers required regulating the authentication scheme and allowing the information retrieval for only legal customers to their services and also they must be competent enough to cancel dishonest customers and disclose their identities.

In recent computing scenarios, thousands of customers can retrieve services from the cloud environment simultaneously. Therefore, the verification procedure of information retrieval must be as efficient as possible with minimal computational complexity. We present a new privacy-enhancing security method for cloud offerings that provide group signatures-based anonymous access. The purpose of the proposed solution is to deliver a required level of security and customer privacy without compromising performance or efficiency. PESS also offers the data integrity and confidentiality of transferred information among clients and cloud providers.

Also, we employ this method as a proof-of-concept torelate the efficiency of PESS with other existing approaches. The results reveal that PESS provides better performance than the other cryptographic methods.

 DOI : 10.18231/2454-9150.2018.0493

The reminder section of the article is structured as follows: We review some prior investigations which match our analysis in Section II. We examine the cryptography based privacy enhancing methods used in the cloud in section III. We discuss the fundamental concepts of group signatures in section IV. In section V we discuss the present PESS and we introduce a new cryptography based privacy enhancing method for retrieving cloud services in section VI. An evaluation process is presented in Section VII. Then, we present our conclusion in section VIII.

II. RELATED WORK

Recently, many research efforts have been devoted to handling security concerns in the cloud environment but only a few studies concerns with customer privacy. Chen and Sion [2] investigate the overhead of common cryptographic primitives and their feasibility for secure cloud services. They propose an encryption technique for retrieving cloud offerings but do not provide any privacy-preserving authentication procedure. Wang et al. [3] implement a pairing-based signature method to achieve the privacy-enhancing access using the Third Party Auditor (TPA). This method employs group verification to decrease communication complexity on the service provider and computation overhead on TPA additionally they [4] present the verification protocols that can deal with dynamic information. They investigate the issues of delivering concurrent public audit ability and information dynamics for data integrity with sufficient privacy. Even though, proposed solutions in [3] and [4] offer a privacy-preserving verification they do not mention anonymous access.

Laurikainen[5] presents the requirements for a protected and anonymous access from cloud computing. However, the author does not provide any solution based on cryptographic method. One more non-cryptographic technique certifying customer confidentiality is investigated by Mowbray and S.Pearson [6]. They present a customer-based privacy manager to decrease the possibility of the exposure of customer data. Hernandez-Ramirez et al. [7] implement a non-cryptographic method to achieve the reimbursements of the shared storage in the cloud without revealing the identity of the customer. This scheme implements Information Dispersal Algorithm (IDA). Still, the techniques used in [5], [6] and [7] do not provide enhanced security against the like ability of customer sessions which lead illegal customer profiling. A ring and GSM-based anonymous and accountable information retrieval scheme is proposed by Jensen et al. [8]. However, implementation of GSM [9] is in effective as the signature size increases with the number of clients.

Angin et al. [10] use a security scheme that employs zero-knowledge proofs (ZKP) for delivering anonymous access. Since this work implements Fiat and Shamir identification method [11] there is an increased communication complexity in client and server sides. Blanton [12] implements Camenisch and Lysyanskaya signature (CLS) method [13] and ZKPto realize anonymous authentication to retrieve cloud services such as music collections, digital libraries, digital newspapers, etc. Lu et al. [14] use a cryptographic method to guarantee anonymity and confidentiality of SPI in the cloud. Chow et al. [15] deal with anonymous access as well as unlink ability in the cloud environment using GSM [16]. In the following section, we investigate the communication complexity of the solutions in [12], [14] and [15].

III. EXISTING CRYPTOGRAPHY BASED PRIVACY ENHANCING METHODS EXPLOITED IN CLOUD SERVICES

In this section, we analyze various modern cryptographic methods which deliver the secret authentication to cloud offerings. The major goal of this study is to provide a cryptography based privacy enhancing scheme for anonymous access to cloud services. In order to measure communication overhead of various solutions, we consider only expensive manipulations like multiplication (mu), exponentiation (ex) and bilinear pairings (bp). As stated by the solutions in [17] and [18], we ignore the fast operations such ashash functions, addition, and subtraction which have a negligible influence on the total performance.

Table 1 displays the comparative study of communication overheads associated with the Blantom method [12], the Lu method [14], the Chow et al. method [15] and our proposed method explained in the subsequent section. Blantom suggests a solution based on CLS [12]. In order to achieve anonymous access, the CLS is used with ZKP protocols. The communication overhead of the Blantom method is variable and mainly hinges on the subscription type. A pairing-based cryptographic method is presented by Lu et al. [14] for guaranteeing anonymous access of customers to retrieve the cloud offerings. Each customer has to sign a message sent by the cloud server and then the customer redirects it back for verification. Chow et al. [15] implement GSM which is based on the methods proposed by Boyen and Waters in [16] and [19] respectively. Chow method is implemented to create a group signature that delivers anonymous access to customers.

Method	Communication Complexity (Elements)	Authentication			Verification		
		mu	ex	bp	mu	ex	bp
Blantom Method [12]	Variable	12	31	30	5	17	6
Lu Method [14]	5	10	14	-	2	1	6
Chow Method [15]	6	15	14	-	6	1	6
Proposed Solution	12	8	10	-	6	12	-

Table 1.Comparative study of the overhead of various methods

 DOI : 10.18231/2454-9150.2018.0493

IV. GROUP SIGNATURES IN CLOUD SERVICES

Group signatures are implemented in several privacy improving methods which are used in cloud computing. Chaum and Heyst [8] firstly proposed GSM in 1991. This method permit each member of a particular group can sign a message using a particular secret key $S_{key}[i]$. The group manager (GM) generates and distributes this key to all the group members. The legitimacy of the sign is checked by the verifier using public key $[P_{key}]$ and it is competent enough to decide whether the signer is really associated with the group without revealing their identity. The identities of the members are visible only in some situations such as violating the rules provided by the service provider. Access right cancellation can be achieved by either a group manager or a revocation manager (RM) who retains the master secret key MS_{key}.

The GSM typically consists of four entities based on the participation. The first entity is the group manager which is responsible to construct a group. Also, it creates and distributes the key S_{key}. The second one is the revocation manager who reveals the identity of malicious users. The third one is a Customer as a group member who preserves $S_{key}[i]$. The customer can sign a message for its group. The fourth one is a verifier which verifies the authority of the sign by applying P_{key}.

Recently, there are several alternatives of GSM are introduced which vary primarily in their characteristics like security, efficiency, the size of the signature and the level of anonymity etc. Group signature can be considered as an important element of access method that encompasses a single attribute indicating the relationship in a group. It has the following characteristics as a merely a legal customer can generate a correct signature for the group. Every member can be traced by GM or RM by the messages signed by members. The verifier is unable to identify the identity of a signer. Other members and a verifier are unable to connect two signatures which have been signed by one customer. Even an invader gains a correct signature, P_{key} and all $S_{key}[i]$ keys he is unable to identify the identity of a signer. A subgroup member is unable to generate signatures. Even GM is unable to generate a correct signature. Each valid sign of the customer has to be constantly recognized through the checking process. If a customer's access is revoked his ability to generate the signatures is restricted instantly. No two members can have same $_{key}[i]$.

V. OVERVIEW OF THE PROPOSED SOLUTION

This chapter illustrates our new privacy-enhancing security solution for the cloud environment. Fig.1 demonstrates the outline of our proposed model.

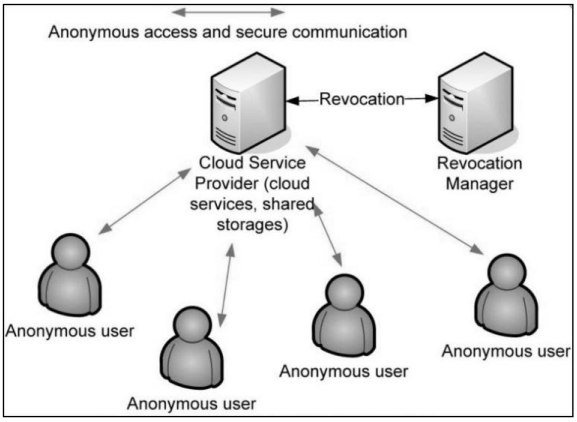

Fig.1 Outline of the proposed model

5.1 Outline Of The Proposed Model

There are three major entities associated with our system as given below:

1. Cloud Service provider or Utility Computing Provider (UCP): UCP manages shared storages and provides the cloud services to the customers. It is usually an organization which acts as a partially trusted party. It delivers services, validates customers while they seeking access to the data. It also provides access attributes to customers. However, when UCP essentials to revoke and detect a dishonest customer then it needs to cooperate with RM.
2. RM: It is also a partially trusted party like government agencies which verifies whether the cancellation of a customer's right is correct or not. Only the collaboration among UCP and RM can disclose the customer identity during the registration process when the customer's access attributes are delivered.
3. Customer (C): C is a normal client who consumes the cloud facilities provided by UCP. Customers are anonymous if they correctly obey the instructions of UCP. In order to improve security, customers employ tamper-resistant memory.

5.2 Security Demands

PESS meets the following security demands:

1. Anonymity: Each authentic customer keeps anonymous while retrieves cloud offerings. Customer identities are concealed if they perform reliably and obey the rules provided by the service provider.
2. Integrity: The information transmitted in the customer's session can be altered only using a secret session key.
3. Confidentiality: Each customer's session to UCP is confidential. Only, a customer having a secret session key able to get information transferred amongst client and UCP.
4. Untraceability: Other customers are not able to track the customer's access and communication records.

 DOI : 10.18231/2454-9150.2018.0493

5. Unlinkability: Normally, the customer's sessions are unlikable. Nobody, expect UCP cooperation with revocation manager, is capable to connect two or more sessions amongst a particular client and UCP.

6. Revocation: Each dishonest customer can be revoked by the cooperation of UCP and RM.

5.3 Cryptographic Methods And Protocols Used In PESS

PESS is implemented using discrete logarithm commitments explained in the work [20]. In this research, the discrete logarithm commitments are converted into a GSM mode. Furthermore, the proposed scheme uses Σ-protocols [21] to validate the discrete logarithm knowledge, representation and equivalence [22]. The Okamoto-Uchiyama Trapdoor One-Way Function which is explained in [23] is used to revoke an illegal customer. To know further details about the implemented scheme refer earlier works presented in [20] and [24].

VI. CRYPTOGRAPHY BASED PRIVACY ENHANCING SCHEME

Our cryptography based privacy enhancing scheme comprises of five different stages: initialization, registration, authentication, secure communication and revocation. Fig.2 describes the fundamental concept PESS.

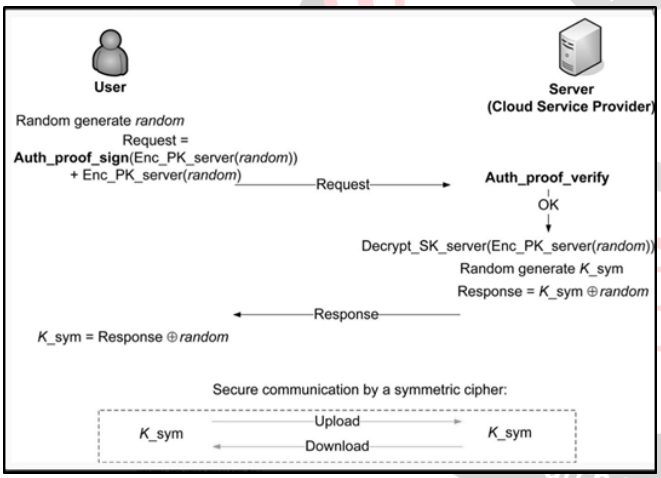

Fig. 2 The working mechanism of PESS.

6.1 Initialization

UCP and RM are responsible to run the initialization phase. The UCP creates a group R which is denoted by a large prime modulus p (1024 bit length), generators r_1, r_2 of prime order q and q/p -1. UCP creates a key pair and stores its private key K_{UCP}. The revocation manager creates a group S which is also a large modulus

$$n = a^2\mathrm{b}$$

Where $a=2a'+1$, $b=2b'+1$ and a, a', b, b' are large primes. The generator $s_1 \in_R \mathbb{Z}^*_n$ is created by RM of the order

$$order(s_1 mod\ a^2) = a(a-1)\ in\ \mathbb{Z}^*_{a_2} \quad \text{and}$$

$$order(s_1) = aa'b'\ \ in\ \ \mathbb{Z}^*_n$$

Revocation manager arbitrarily selects secret values b_1, b_2, b_3. It calculates authentication proof AU_{proof} as follows

$$AU_{proof} = {s_1}^{b_1}\ mod\ n$$

AU_{proof} is common for all entities in the system and can estimate other proof such as ${AU_{proof}}^1$, ${AU_{proof}}^2$,… ${AU_{proof}}^n$. These proofs are calculated from $b_1^1, b_1^2, \ldots . b_1^n$ that are associated with various customer rights to access data from the cloud. As a final point, revocation manager estimates s_2 and s_3 as

$$s_2 = {s_1}^{b_2} mod\ n \text{ and } s_3 = {s_1}^{b_3} mod\ n$$

RM stores secret values a and b as revocation key K_{Rev}. All factors p, q, n, s_1, s_2, s_3, r_1, r_2, AU_{proof} are disseminated.

6.2 Registration

In this process, the customer registers and requests a customer master key that is used for anonymous authentication. Initially, customers must register on UCP. Then, UCP verifies the customer's ID. After that, the customer creates secret values α_1, α_2 and generates the commitment as

$$C_{UCP} = r_1^{\alpha_1} r_2^{\alpha_2}\ mod\ p$$

then, the customer digitally signs C_{UCP}, and transmits this signature $sign_c(C_{UCP})$with the correctness proof to UCP by the notion of Camenisch and Stadler [22]. The correctness proof is calculated as

$$P_{Key}(\alpha_1, \alpha_2 : C_{UCP} = r_1^{\alpha_1} r_2^{\alpha_2}\ mod\ p)$$

UCP verifies the proof and the signature. Next, UCP stores the pair C_{UCP}, $sign_c(C_{UCP})$. After that, UCP signs the commitment $sign_{SP}(C_{UCP})$and transmits to the customer.The customer requests a master key from revocation manager. The customer calculates AU'_{proof} such that

$$AU'_{proof} = {s_1}^{\alpha_1} {s_2}^{\alpha_2} mod\ n$$

and transmits it with C_{UCP}, $sign_{SP}(C_{UCP})$ and the creation of correctness proof to revocation manager. The creation of correctness proof is as follows

$$P_{Key}(\alpha_1, \alpha_2 : C_{UCP}) = r_1^{\alpha_1} r_2^{\alpha_2}\ mod\ p\ \wedge\ AU'_{proof} = {s_1}^{\alpha_1} {s_2}^{\alpha_2}$$

RM verifies the proof, UCP's signature $sign_{UCP}(C_{UCP})$ and calculates a secret influenceα_{RM} as

$$AU'_{proof} = {s_1}^{\alpha_1} {s_2}^{\alpha_2} {s_3}^{\alpha_{RM}}$$

then, customer gains own customer master key K_{mast} which is a triplet$(\alpha_1, \alpha_2, \ldots\ldots \alpha_{RM})$. The customer obtains α_{RM} only by collaboration with revocation manager since RM only can find the factors n. The secret values α_1, α_2 is not disclosed to a customer as they are kept in a protected tamper-resistant device to avert the collusion attack. This memory should be also secured against side-channel attacks, as stated in [25]. Also, customers cannot generate their master key since only revocation manager knows K_{Rev}. Allun revoked customer can

 DOI : 10.18231/2454-9150.2018.0493

repeat the call for K_{mast} or request other AU_{proof} if UCP accepts the request.

6.3 Authentication

In access phase, the i[th] customer C_i anonymously accesses UCP. Anonymous access involves two messages exploited to validate C_i and generate a secret key. C_i creates random value *rand* such that

$$rand \in_R = \{0,1\}^{\lambda}$$

Here,λ represents the length of a published secret key.C_i encrypts *rand* by P_{Key}. The encrypted message $Enc\_P_{Key}\_serv(rand)$ is signed by the algorithm, namely Auth_proof_sign in the GSM mode which guarantees anonymous access. We assume that the factors p, q, n, s_1, s_2, s_3, r_1, r_2 and $AU'_{proof} = {s_1}^{\alpha_1}{s_2}^{\alpha_2}{s_3}^{\alpha_{RM}}$ are shared and H is a secure hash function. Each C_i executes the following Auth_proof_sign algorithm to validate the *sign(rand) and* secret key.

$$K_{sess} = \in_R \{0,1\}^{\lambda}$$

$$AU = AU_{Proof}^{K_{sess}} \bmod n$$

$$e_1 = s_3^{K_{sess} \propto RM} \bmod n$$

$$e_2 = s_3^{K_{sess}} \bmod n$$

$$a_1, a_2 \in_R \{0,1\}^{m+k+3\lambda}$$

$$a_3 \in_R \{0,1\}^{m+k+4.5\lambda}$$

$$a_{sess} \in_R \{0,1\}^{m+k+\lambda}$$

$$\overline{AU_{proof}} = s_1^{a1} s_2^{a2} s_3^{a3} \bmod n$$

$$\overline{AU} = AU_{Proof}^{a_{sess}} \bmod n$$

$$\bar{e_1} = s_3^{a3} \bmod n$$

$$\bar{e_2} = s_3^{a_{sess}} \bmod n$$

$$e = H(Enc\_P_{key}\_serv(rand, AU, \overline{AU}, \overline{AU_{proof}}, e_1, e_2, \bar{e_1}, \bar{e_2})$$

$$x_1 = a_1 - eK_{sess}\alpha_1$$

$$x_2 = a_2 - eK_{sess}\alpha_2$$

$$x_3 = a_3 - eK_{sess}\alpha_{RM}$$

$$x_{sess} = a_{sess} - eK_{sess}$$

As a final point, the signature elements AU, $\overline{AU}$, $\overline{AU_{proof}}$, e_1 , e_2 , $\bar{e_1}$, $\bar{e_2}$, x_1, x_2, x_3, x_{sess}, $Enc_P_{key}_serv(rand)$are directed to UCP as a reply. UCP checks the signed request message that contains $Enc\_P_{key}\_serv(rand)$, AU, $\overline{AU}$, $\overline{AU_{proof}}$, e_1 , e_2 , $\bar{e_1}$, $\bar{e_2}$, x_1, x_2, x_3, x_{sess}. Now, UCP executes the following Auth_proof_verify algorithm:

$$e_1 \not\equiv {e_2}^{rev} \bmod n$$

$$\overline{AU_{proof}} \equiv AU^e {s_1}^{x_1}{s_2}^{x_2}{s_3}^{x_3} \bmod n$$

$$\overline{AU} \equiv AU^e AU_{proof}^{x_{sess}} \bmod n$$

$$\bar{e_1} \equiv {e_1}^e {s_3}^{x_3} \bmod n$$

$$\bar{e_2} \equiv {e_2}^e {s_3}^{x_{sess}} \bmod n$$

If above constraints are satisfied then UCP continues in the subsequent phase. Else, UCP halts the execution. UCP decodes $Enc\_P_{Key}\_serv(rand)$ using its private key to determine *rand*. Then, UCP creates *K_sym* and perform XOR operation of *K_sym* with *rand.* Now, UCP transmits (*rand* ⊕ K_sym) toC_i as a response.

6.4 Secure Communication

The customer C_i can access information from UCP if and only if the anonymous authentication process is successful. Data integrity as well as confidentiality is achieved using the symmetric cipher. In order to support various types of platforms, we recommend using AES which is renowned cryptography method. For data encryption and decryption, all the customers and UCP employ *K_sym* generated in the authentication process.

6.5 Revocation

If customers violate the rules provided by the service provider, they get revoked immediately by the revocation manager.

Initially, RM calculates the arbitrary session key K_{sess} using e_2 and the secret impact value α_{RM} using e_1. Then, it stores α_{RM} in a blacklist. If the customer employs revoked key and the condition $e_1 \equiv e_2\ \alpha_{RM}$ holds then the customer access is barred. If a customer violates the rules of UCP, he/she can be recognized by the cooperation of RM and UCP. Initially, RM obtains α_{RM} from the distrusted session established by UCP. Next, RM calculates the related C_{UCP} in the database. If UCP delivers the explicit sign of the customer's breach to RM, then the revocation manager sends C_{UCP} to UCP. Now, UCP is capable of disclosing the identity of a customer but only with the cooperation of revocation manager.

VII. EXPERIMENTALANALYSIS

7.1 Performance Analysis of PESS

In this study, our PESS is realized using JAVA. We assume that the customer's end node devices with enough computational capacity like a smart phone, tablet, laptop, or personal computer. In contrast, we consider UCP possesses servers with adequate processing power to guarantee hundreds of sessions with the customer. Our proposed solution is tested on Intel®Xeon®Quad-core ProcessorX3440 (8 MB cache, 2.53GHz) with 4 GB RAM. In the anonymous authentication process of PESS, a customer communicates with UCP. The computation procedure performed by UCP is denoted as the verification phase. The total time consumed for authentication as well as verification phases ismeasured.

 DOI : 10.18231/2454-9150.2018.0493

Size of Black List	Total Time [ms]		
	Session 1	Session 2	Session 3
rev = 0	54	546	1042
rev = 10	70	721	1272
rev = 15	106	920	1891
rev = 25	129	1227	2477
rev = 35	145	1439	2668
rev = 45	164	1627	3139
rev = 55	183	1818	3402
rev = 65	193	2071	3881
rev = 75	211	2224	4275
rev = 85	232	2394	4510
rev = 95	258	2571	4751
rev = 100	279	2824	5103

Table 2 Performance Evaluation of PESS

The total time given in the Table 2 displays two different circumstances: (i) with a blank black list (*i.e., rev = 0*) and (ii) with the black list that contains the number of revocation as 10, 15, 25….100. Fig.3 shows the impact of the size of the blacklist on execution time.

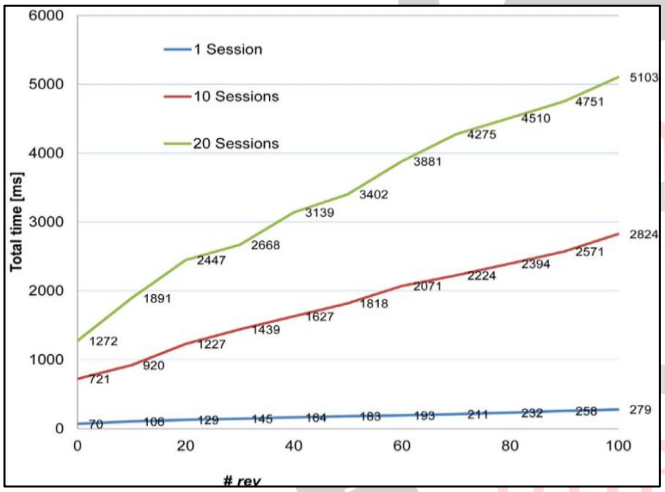

Fig.3 The impact of the length of the blacklist on the execution time

7.2 Comparison of computational overhead with Related Works

To prove the effectiveness of PESS, we relate the computational overhead of our anonymous authentication process with other related solutions: Blantom method [12], Lu method [14] and Chow method [15]. Initially, we compare the authentication phase that executes on the customer side. In the authentication phase, Lu method takes 10 mu + 14 ex, Chow method takes 15 mu + 14 ex. The number of operations in Blantom method is variable one and hinges on the subscription type. The authentication phase of PESS takes only 5 mu+ 8 ex and is the most competent from other compared methods.

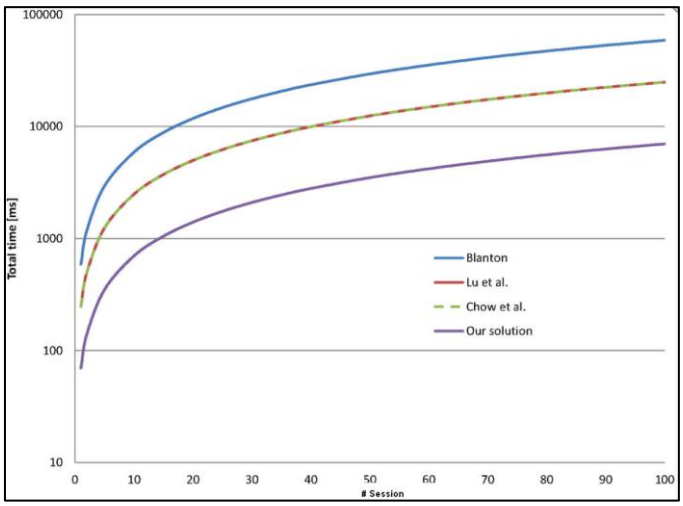

Fig. 4 Comparative analysis of computational overhead of PESS with the other related solution

In verification phase, Lu method, Chow method and Blantom method take 6 pairing operations. The verification phase of PESS on the UCP side takes only 6mu + 10 ex. We find that PESS has 0 paring operations. The comparative performance analysis of PESS and other methods is depicted in Fig. 4. The verification phase of PESS is more efficient than verification processes in other methods and only consumes 30 % of the total time.

VIII. CONCLUSION

There are several privacy-enhancing schemes used in a cloud computing environment. In this research, we propose a new cryptography based privacy enhancing security solution for the cloud computing environment. A non-bilinear group signature method is adapted to provide the anonymous access. PESS not only delivers customer anonymity and also offers integrity and confidentiality for transmitted data and perform reasonable revocation process for dishonestcustomers. The authentication and verification processes employed in this study is more effective than othermethods on the customer as well as server side by eliminating expensive pairing processes and takes less number of multiplication and exponentiation operations.Therefore, utility computing providers with PESS can validate more customers simultaneously.

REFERENCES

[1] P. Mell and T. Grance, “The NIST Definition of Cloud Computing” , NIST Special Publication 800-145, 2011.

[2] Y. Chen and R. Sion, “On securing untrusted clouds with cryptography”, In Proceedings of the 9th annual ACM workshop on Privacy in the electronic society, New York, ACM, 2010, pp.109–114.

[3] C. Wang, et al., “Privacy-preserving public auditing for data storage security in cloud computing”, in INFOCOM,2010 Proceedings IEEE, San Diego march 2010, pp. 1–9.

 DOI : 10.18231/2454-9150.2018.0493

[4] Q. Wang et al. "Enabling public auditability and data dynamics for storage security in cloud computing", in Parallel and Distributed Systems, IEEE Transactions on, vol. 22, no. 5,IEEE, 2011. pp. 847–859.

[5] R. Laurikainen, "Secure and anonymous communication in the cloud", in Aalto University School of Science and Technology, Department of Computer Science and Engineering, Tech. Rep. TKK-CSE-B10, 2010, pp. 1-5

[6] M. Mowbray and S. Pearson, "A client-based privacy manager for cloud computing", in Proceedings of the Fourth International ICST Conference on COMmunication System softWAre and middle waRE, ser. COMSWARE '09, New York, ACM, 2009, pp. 5:1–5:8.

[7] E.M. Hernandez-Ramirez et al. "A Comparison of Redundancy Techniques for Private and Hybrid Cloud Storage", in JART Journal of Applied Research and Technology, vol. 10, no. 6, pp. 1-9, 2012.

[8] M. Jensen et al., "Towards an anonymous access control and accountability scheme for cloud computing", in Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on, Miami, IEEE. 2010, pp. 540 –541.

[9] D. Chaum and E. Van Heyst, "Group signatures", in Advances in Cryptology EUROCRYPT91. 1991, pp. 257–265.

[10] P. Angin et al., "An entity-centric approach for privacy and identity management in cloud computing", in Reliable Distributed Systems, 201029th IEEE Symposium on, New Delhi, IEEE. 2010, pp. 177–183.

[11] A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems", in Advances in Cryptology-Crypto86. 1987, pp. 186–194.

[12] M. Blanton, "Online subscriptions with anonymous access", in Proceedings of the 2008 ACM symposium on Information, computer and communications security, ser. ASIACCS '08, New York, ACM. 2008, pp. 217–227.

[13] J. Camenisch and A. Lysyanskaya, "Signature schemes and anonymous credentials from bilinear maps." in Advances in Cryptology– CRYPTO2004, 24th Annual International Cryptology Conference, Santa Barbara, California, USA. 2004, pp. 56–72.

[14] R. Lu et al., "Secure provenance: the essential of bread and butter of data forensics in cloud computing", in Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, ser. ASIACCS '10, New York, ACM, 2010, pp. 282–292.

[15] S. Chow et al., "Spice–simple privacy-preserving identity-management for cloud environment", in Applied Cryptography and Network Security. 2012, pp. 526–543.

[16] X. Boyen and B. Waters, "Compact group signatures without random oracles", in Advances in Cryptology EUROCRYPT 2006. 2006, pp. 427–444.

[17] L. Malina and J. Hajny, "Accelerated modular arithmetic for low-performance devices", in Telecommunications and Signal Processing(TSP), 2011 34th International Conference on, Budapest, IEEE. 2011, pp. 131–135.

[18] L. Malina and J. Hajny, "Efficient modular multiplication for programmable smart-cards", in TelSys. Telecommunication Systems, pp.1-8. 2013.

[19] X. Boyen and B. Waters, "Full-domain subgroup hiding and constant-size group signatures", Public Key Cryptography–PKC 2007.Beijing, China. 2007, pp. 1–15.

[20] J. Hajny and L. Malina, "Unlinkable attribute-based credentials with practical revocation on smart-cards", in Proceedings of the 11th international conference on Smart Card Research and Advanced Applications, ser. CARDIS'12. Springer-Verlag, 2013, pp. 62–76.

[21] R. Cramer, "Modular design of secure, yet practical cryptographic protocols", Ph.D. dissertation, University of Amsterdam, 1996.

[22] J. Camenisch and M. Stadler, "Proof systems for general statements about discrete logarithms", Tech. Rep., 1997.

[23] T. Okamoto and S. Uchiyama, "A new public-key crypto system as secure as factoring", in Advances in Cryptology - EUROCRYPT 98, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, vol.1403, pp. 308–318, 1998.

[24] J. Hajny and L. Malina, "Practical revocable anonymous credentials", in Communications and Multimedia Security, Canterbury, UK. 2012, pp. 211–213.

[25] L. Martínez-Ramos et al., "Achieving Identity-Based Cryptography in a Personal Digital Assistant Device", JART. Journal of Applied Research and Technology, vol. 9. no. 3, pp. 1-11, 2011.

 DOI : 10.18231/2454-9150.2018.0493