BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **15**, No 1

Sofia • 2015 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.1515/cait-2015-0005

Privacy Preservation of a Group and Secure Data Storage in Cloud Environment

K. Govinda[1], *E. Sathiyamoorthy*[2]

[1] *School of Computing Science and Engineering, VIT University, Vellore-14, Tamilnadu, India*
[2] *School of Information Technology and Engineering, VIT University, Vellore-14, Tamilnadu, India*
Emails: kgovinda@vit.ac.in esathiyamoorthi@vit.ac.in

Abstract: *Cloud computing has become a victorious archetype for data storage, as well as for computation purposes. Greater than ever it concerns user's privacy, so that data security in a cloud is increasing day by day. Ensuring security and privacy for data organization and query dispensation in the cloud is important for superior and extended uses of cloud based technologies. Cloud users can barely have the full benefits of cloud computing if we can ensure the real user's privacy and his data security concerns this approach along with storing thin-skinned personal information in databases and software spread around the cloud. There are numerous service suppliers in WWW* (*World Wide Web*)*, who can supply each service as a cloud. These cloud services will switch over data with a supplementary cloud, so that when the data is exchanged between the clouds, the problem of confidentiality revelation exists. So the privacy revelation problem concerning a person or a corporation is unavoidably open when releasing or data distributing in the cloud service. Confidentiality is a significant issue in any cloud computing environment. In this paper we propose and implement a mechanism to maintain privacy and secure data storage for group members or a community in cloud environment.*

Keywords: *Cloud computing, privacy, data security, community, GDH.*

1. Introduction

Cloud computing has emerged as a victorious prototype that significantly simplifies the consumption of computing and data storage space infrastructures of both large and small scale enterprises [1]. The growing concerns about data protection and confidentiality in cloud environment appear since the big assessment of vulnerabilities. There are many security flaws found in the services and user data outflow incidents reported for a large amount of cloud based applications. Ensuring data security and users' privacy in data administration and handing out queries submitted to the cloud is therefore crucial for improved and broader uses of the cloud [3]. On the other hand, under the condition that such secure and privacy-preserving data security services are very tough, since safe keeping tribulations can arise in multiple levels of the data services, security and privacy protection may impede functionality and performance of the data services. This tutorial aims to cover some common cloud security and privacy threats and the relevant research, while focusing on the works that protect data confidentiality and query access privacy for sensitive data being stored and queried in the cloud. We afford complete study of the modern schemes and techniques for protecting data confidentiality and accessing privacy, which make dissimilar tradeoffs in the multi perceptual space of security, privacy, functionality and performance [5]. We also identify their boundaries and further discuss future research directions in cloud data security and privacy. The courageous new globe of cloud computing offers numerous profits provided that the solitude and data protection issues can be recognized and successfully pulled down [7]. A very large number of users intend to take advantage of the power of the cloud based environment. This kind of environment has many advantages, such as unlimited elasticity: With admittance to millions of dissimilar pieces of software and databases, and the capability to unite them into personalized services, the users are proved competent to find the targeted solution they need, to contribute to their ideas, and enjoy online games, video, and virtual worlds; improved dependability and safety: the users no longer have to be anxious about their hard drives crashing or their machines being stolen; Superior partnership: By enabling online distribution of the information and cloud based applications in a cloud offers the users new ways of working and playing together; Portability: The users can access their data and tools anywhere they want with cloud connectivity; Simpler policy: With data and the software being stored in the cloud, users do not need a powerful computer [6]. These cloud computing applications can be interfaced using a cell phone, a PDA, a personal video recorder, an online game console, cars, or even sensors built into their clothing.

The most important key feature of this specific proposal is that it uses a trusted party, i.e., an agent who preserves the user's privacy and maintains the data security and integrity. Each time the data are sent to the cloud, they will be passed through the agent who makes remarkable modifications over the data by applying encryption and a signature as requirements [10].

2. Literature review

The survey actually begins with the words said by L. Frank Kenney, research director of Gartner CBS: "The future of cloud computing will be permeated with the notion of brokers negotiating relationships between providers of cloud services and the service customers. In this context, a broker might be software, appliances, platforms or suites of technologies that enhance the base services available through the cloud. The enhancement includes managing access to these services, providing greater security or even creating completely new services” [1].

Although cloud service brokers may be delivered through technology, there is still a need for brokerage businesses to exist in order to take advantage of the brokers. A brokerage is a service in which a broker may simply be B2B technology, and Gartner believes that Cloud Service Brokerages (CSBs) are one of the most necessary and attainable opportunities for cloud service providers. CSBs will arrange the relationships between a service consumer and a service provider.

2.1. GDH algorithm

GDH (Group Diffie-Hellman) is a key distribution algorithm that helps multi users in a team to share a secret key between themselves without the need to exchange the exact key as shown in Fig. 1. The power and beauty of the GDH algorithm lies in the fact that multiplication is commutative. An overview of the algorithm is given below.

To share a secret key, the users U1, U2, U3 and U4 agree on public numeric constants p and g [6]. Here p is any random prime number and g is the generator of p which is less than p. The users U1, U2, U3 and U4 select random numbers r_1, r_2, r_3, and r_4 by their own respectively.

- Now User U1 picks the generator g and computes g^{r_1} and sends it to U2.
- The User U2 computes g^{r_2}, then with the received g^{r_1} computes $g^{r_1} \times g^{r_2}$, and after that the set consisting of g^{r_1}, g^{r_2}, $g^{r_1} \times g^{r_2}$ is sent to the User U3.
- The User U3 with the computed g^{r_3}, computes $g^{r_1} \times g^{r_2} \times g^{r_3}$ and sends g^{r_1}, g^{r_2}, g^{r_3} and $g^{r_1} \times g^{r_2} \times g^{r_3}$ to User U4.
- Now the User U4 is waiting with g^{r4} to receive this package and computes
- $g^{r_1} \times g^{r_2} \times g^{r_3} \times g^{r_4}$.
- At the end every User gets the real group key.
- The Final User shares the partial keys to the other users without their respective partial key. For example, the User U1 will receive the keys g^{r_2}, g^{r_3}, g^{r_4} except g^{r_1} which he has already got.

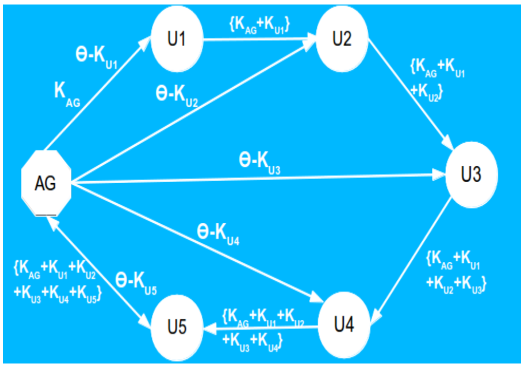

Fig. 1. Key exchange in a group

2.2. RSA algorithm

RSA (Rivest-Shamir-Adleman) algorithm is a public key algorithm developed by Rivest, Adi Shamir and Len Adleman that is used for encryption, decryption, signature and key agreement. RSA uses typically keys of the size from 1024 up to 2048. Overviews of RSA algorithms are given below [9]. As in any data encryption algorithm, RSA has its own powerful mathematical background that enables RSA to be the strongest procedure for both data encryption, as well as digital signature.

2.3. Key generation

- Select two large prime numbers p and q.
- Compute $n = pq$. The computed n is made public.
- Now compute $\Phi(n) = (p-1)(q-1)$.
- Choose a random number e as a public key within the range $0<e<\Phi(n)$, such that GCD $(e, \Phi(n)) = 1$.
- Find private key d is such that $d = (e-1) \bmod \Phi(n)$.

2.4. Encryption

- Consider the device A that needs to send a message to B in a secured manner using RSA algorithm.
- Now e is B’s public key. Since e is public, A is allowed access to e.
- For encryption of the message M of A which is within the range $0<M<n$ is converted to a cipher.
- The cipher text $C = M^e \bmod n$.

2.5. Decryption

- Now the cipher text C is sent to B from A.
- Device B calculates the message with its private key d, where message $M = C^d \bmod n$.

2.6. Digital Signature algorithm

- Digital Signature method is a simple method that is used to recognize the sender of the message. Using RSA algorithm, this can be simply implemented by inversing the key usage, i.e., encrypting the data with the private key and decrypting the data with a public key [8].The signature is generated by the group key [9].
- Using the above specified procedures and algorithms, the proposed methodology uses them to make itself appropriate for the demanding requirements of privacy preservation and data security [7].

3. The method proposed

There are many identity based privacy preservation issues that everybody should concentrate on in cloud computing environment: The first and most important issue is the revelation of thin-skinned private information when the exchanging data occur between the cloud service inside the cloud environment, and this thin-skinned personal information includes: Personally identifiable information, Usage data, Unique device identities and so on. The next visible issue is that people are getting unsuitable or unofficial access to personal data in the cloud by taking benefit of certain vulnerabilities, such as lack of access control enforcement, security holes and so on. The third problem is due to the fact that cloud computing is highly dynamic, because the service communications can be created in a more self-motivated way than traditional e-Commerce scenarios. Services can potentially be aggregated and changed dynamically by service providers and revolutionize the provisioning of services. In such scenarios, private sensitive data may move around within an association or across governmental restrictions, so that tolerable fortification of this information must be maintained even though there are changes. So to design a method to protect the privacy in cloud computing must meet the dynamical exchange of data. In our current proposal as we have earlier discussed, we use as a trusted party an agent who preserves our privacy and maintains data security and integrity. This outline idea can also be simply described as shown in Fig. 2.

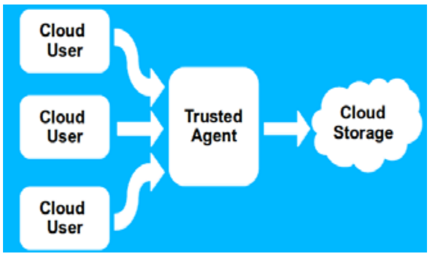

Fig. 2. Proposed architecture

As described above the communication between the user and the cloud occurs only through the trusted agent. Whenever the communication is established between the user and the agent, or between the agent to the cloud service provider, the system follows a simple protocol explained below.

The first phase of the protocol is the key generation and distribution phase. This phase starts by sharing users' details with the trusted agent and then each user of the trusted agent shares a unique key among themselves, using the GDHA. The agent will act as the terminal user and will split the key and distribute it to the users connected to the cloud through him. At the end all the users are assigned with the key that will help the agent to identify the user.

In the same method the trusted agent and the cloud manager share a secret ID for authentication between them, using the same DH algorithm. This helps the cloud manager to recognize the agent. Thus the keys for authentication are shared.

The current phase continues further by a key set generation with respect to RSA encryption and digital signature standards. The user, the trusted agent and the cloud manager will generate their own public key, as well as the private key set. Now the user and the trusted agent share the public key and the same occurs on the other side for the trusted agent and the cloud manager. Thus all the keys are generated and distributed as described in phase 1. A simple diagram spotting the keys used in the protocol is shown below.

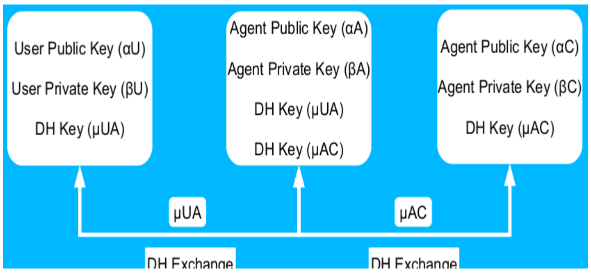

Fig. 3. Key exchange between the user, the agent and the cloud provider

Phase 2 of the proposal is the data storage phase. When the user wants to store the data in the cloud, the following procedure is followed. The user encrypts the data with his public key and computes the message digest for the encrypted text using the MD5 algorithm. The User also makes another attachment that consists of the signed ID that is further encrypted with the agent's public key. This package is now sent to the trusted agent.

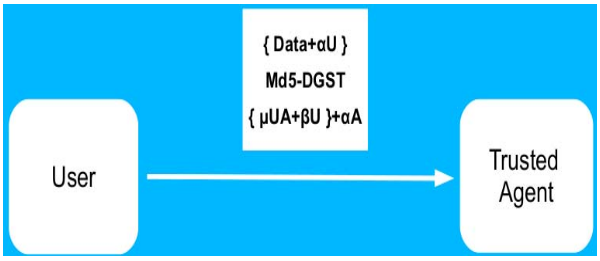

Fig. 4. Data package from the user to the trusted agent

The trusted agent now receives the package. The agent removes the attachment and decrypts it and then un-signs with the user's public key in order to recognize the user with the user ID. On successful authentication of the user, the data part is verified for integrity by computing MD5 hash for the encrypted data with a positive result of the data integrity check. The agent makes a log of the user data incoming with the time stamp. Now the user signature is removed and the new attachment contains the encrypted data and agent's signature, signed with his private key, which is sent to the cloud manager. After the verification process by the cloud manager, the data will be stored in the cloud and through this mechanism the privacy of the group members is maintained.

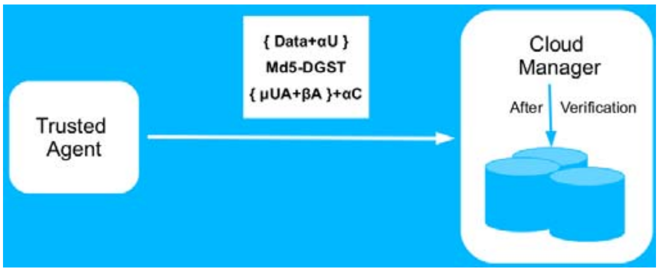

Fig. 5. Data package sent to the cloud for storage

The cloud manager receives the package. The package is further authenticated after decrypting the attachment with the cloud manager’s private key and unsigned with the agent’s public key. At successful result the integrity of the encrypted data is verified once again by MD5, then the package is stored in the cloud. After that the stored successful log is sent to the agent after encrypting with the agent’s public key. The agent receives the message and knows the result with the help of his private key; in this way the privacy and integrity of the data is maintained.

4. Implementation

The mechanism above specified is implemented in a HPC (High Performance Computing) Infrastructure consisting of 6 racks powered by AMD Opteron. Each rack consists of two cores, which in turn have six processors. The implementation is carried out on Ubuntu Linux platform, enabling an open stack cloud environment. Among all, two cores were dedicated to the Trusted Manager. Four storage nodes were created in a distributed manner. Ten client nodes were created and established in the deployed system. The keys for data encryption were generated using the python programs and high level keys are built using open SSL framework and libraries.

5. Result analysis

Table 1 is a sample of the key sets generated by the trusted agent. In a similar way, keys of multiple, higher bit sizes are also generated and distributed in and around the cloud, based on the request and its priority. The graph in Fig. 6 generated below determines the time taken to generate keys with respect to RSA asymmetric crypto-system with variable key sizes. The graphs in Figs 7 and 8 given below clearly define the encryption and decryption time which varies with respect to different key sizes.

Table 1. Sample key sets generated with respect to RSA algorithm

p	Q	pQ	$(p–1)\ (Q–1)$	$e(p_{\mathrm{U}}$ key)	$d(p_{\mathrm{r}}$ key)
1236553	1236583	1529100418399	1529097945264	1235141	$8.09624164366659\times10^{-007}$
1236611	1236623	1529221604653	1529219131420	1235149	$8.09618920470324\times10^{-007}$
1236629	1236643	1529268596447	1529266123176	1235159	$8.09612365695429\times10^{-007}$
1236659	1236661	1529327955599	1529325482280	1235167	$8.0960712195193\times10^{-007}$
1236667	1236701	1529387315567	1529384842200	1235177	$8.09600567368078\times10^{-007}$
1236709	1236713	1529454097517	1529451624096	1235183	$8.09596634668709\times10^{-007}$
1236727	1236737	1529506039799	1529503566336	1235191	$8.09591391128983\times10^{-007}$
1236743	1236751	1529543141993	1529540668500	1235239	$8.09559931316936\times10^{-007}$
1236757	1236761	1529572824077	1529570350560	1235243	$8.09557309776295\times10^{-007}$
1236769	1236787	1529619821203	1529617347648	1235249	$8.09553377497169\times10^{-007}$
1236791	1236797	1529659398427	1529656924840	1235251	$8.09552066745949\times10^{-007}$
1236803	1236811	1529691555233	1529689081620	1235263	$8.09544202327763\times10^{-007}$
1236827	1236857	1529778132739	1529775659056	1235281	$8.09532405986978\times10^{-007}$
1236883	1236901	1529901819583	1529899345800	1235287	$8.09528473949779\times10^{-007}$
1236953	1236959	1530060145927	1530057672016	1235303	$8.09517988703986\times10^{-007}$

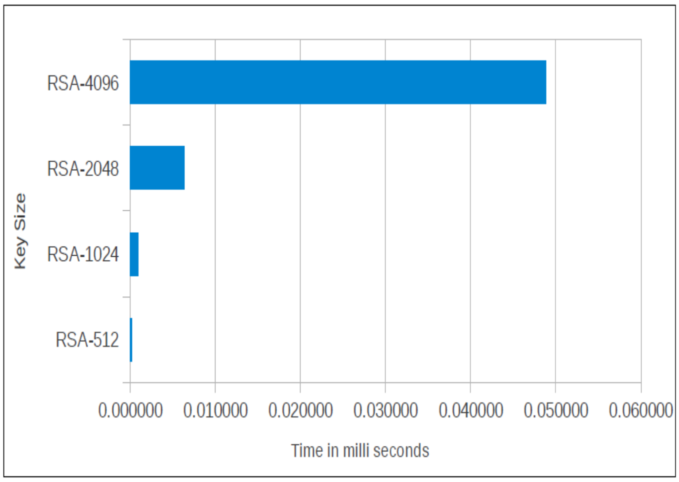

Fig. 6. Key generation

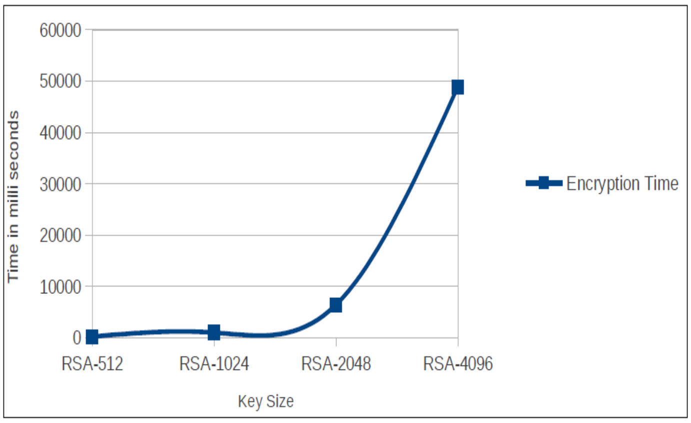

Fig. 7. Encryption vs key size

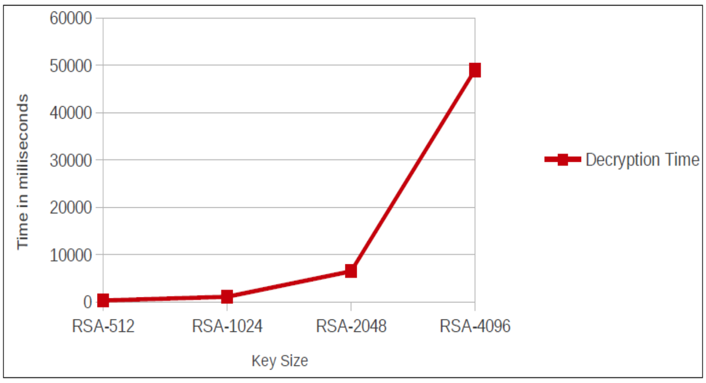

Fig. 8. Decryption vs key size

6. Conclussion

The advantage of the system proposed is that it is highly suitable for dynamic environment with the security assurance and privacy preservation required. The suggested system is of combatable multi-level architecture, so it is easy to transform it into a Platform as a Service (PaaS) in a highly distributed cloud computing environment. The protocol can be further enhanced by modifying the algorithms that are suitable for the environment or with respect to the applications. Moreover, this work can be implemented for some specific applications.

References

1. Santos, N., K. P. Gummadi, R. Rodrigues. Towards Trusted Cloud Computing. – In: USENIX HotCloud, 2009.
2. Agrawal, Rakesh, Ramakrishnan Srikant. Privacy-Preserving Data Mining. – In: Proc. of ACM Management of Data (SIGMOD), 2000, pp. 439-450.
3. Aggarwal, C. C. On k-Anonymity and the Curse of Dimensionality. – In: VLDB, 2005, pp. 901-909.
4. Casassa-Mont, M., S. Pearson, P. Bramhall. Towards Accountable Management of Ldentity and Privacy: Sticky Policies and Enforceable Tracing Services. – In: Proc. DEXA’2003, 2003, pp. 377-382.
5. Rivest, R.L., A. Shamir, L. Adleman. A Method for Obtaining Digital. Signatures and Public-Key Cryptosystems. The Technical Paper to Laboratory for Computer Science, Massachuset Institute of Tech,Cambridge.
6. Hao, Yong, Yu Cheng, Kui Ren, Distributed Key Management with Protection against RSU Compromise in Group Signature Based VANETs. 3rd Ed. Vol. **2**. IEEE BLOBECOM 2008.
7. Kaliski, Burt. The Mathematics of the RSA Public-Key Cryptosystem. – In: RSA Laboratories. P. Kitsos, N. Sklavos and O. Koufopavlou, Eds. An Efficient Implementation of the Digital Signature Algorithm, IEEE, 2008.
8. Diffe, W., M. Hellman. New Directions in Cryptography. – IEEE Trans. Inform. Theory IT-22, November 1976, pp. 644-654.
9. Liu, G. M. Very Simple Schemes for Group Signatures. Master Thesis, Chung Yuan Christian University, June 2003.
10. Chaum, D., E. van Heyst. Group Signatures. – In: Andances in Cryptology EUROCRYPT’91, Vol. **547**, 1991, pp. 257-265.