www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 4 Issue 8 Aug 2015, Page No. 13832-13837

Building Secure, Concurrent, Distributed DBaaS with Confidential Data over Threshold Access Structure

A.Blessy[1], Varun Chand[2]

[1]Freelancing in Information Security related projects,
Tirunelveli, Tamilnadu, India
blessy2789@gmail.com

[2]Assistant Professor, Department of Information Technology,
College of Engineering and Management, Punnapra, Alappuzha, Kerala, India
varunchand123@gmail.com

Abstract: ***In a cloud, Guaranteeing confidentiality within the Database as a service (DBaaS) paradigm remains a problem. Here, we propose ConfidentialConcurrenttoSecureDBaas with some modifications to produce availability, security, accessibility and reliability without exposing unencrypted information to the cloud provider. It additionally permits multiple, freelance and regionally distributed clients to execute synchronal operations on encrypted data by eliminating the need for intermediate server between the cloud consumer and also the cloud provider; to preserve data confidentiality and consistency at the client and cloud level; To achieve this, ConfidentialConcurrentAccessToDBaaS integrates existing cryptographic schemes, isolation mechanisms, and novel strategies for management of encrypted metadata on the untrusted cloud database.***

Keywords: Confidential Concurrent to Secure DBaaS, Database as a Service, Confidential Concurrent Access To DBaaS

1. Introduction

Today for many organizations they need to store their enormous amount of data. Among these, cloud computing is the most cost effective and flexible network storage providers but it has some security issues. Cloud computing provide accuracy, so more data can be centralized into the clouds. Users of this technology are relieved from the data storage and maintenance as they entrust their valuable data in to the clouds. The most important security concerns in cloud are the data security, data confidentiality, integrity, availability, accessibility privacy due to internet based data storage and management. For an organization the extremely important asset is the data. If the data is disclosed the enterprise users will face serious issues from their business competitors and the public. Along with data confidentiality, scalable and flexible access control is also desired by the cloud users. Traditionally, the sensitive data is encrypted and stored on the servers and the decryption keys are disclosed only to the authorized users. It also lacks in flexibility and scalability. This paper focuses on the survey of different encryption schemes and is given in the following sections. Section II presents the literature survey of different encryption schemes and a comparison table and section III designs the proposed work and section IV concludes with discussions.

2. Literature Survey

In cloud computing, there are different existing schemes that provide security, data confidentiality and access control. Users need to share sensitive objects with others based on the recipients' ability to satisfy a policy in distributed systems.

2.1 Balancing Confidentiality and Efficiency in Untrusted Relational DBMS

Hash-based method for database encryption is proposed. Indexing information attached to the encrypted database which can be used by the server to select the data from database. Here Server can select a data to be returned in response to a query without the need of disclosing the database content. But it is only suitable for selected queries.

2.2 Supporting Security and Consistency for Cloud Database

Architecture that avoids any intermediary component, thus achieving availability and scalability comparable to that of unencrypted cloud database services. Guarantees data consistency in scenarios in which independent clients concurrently execute SQL queries, and the structure of the database can be modified. Reduced isolation levels for multi-version systems have never been characterized before despite being implemented in several products. Concurrent modifications of the database structure are supported but at the price of higher overhead and stricter transaction isolation levels.

2.3 An Integrated Experimental Environment for Distributed Systems and Networks

Netbed, a descendant of Emulab, provides an experimentation facility that integrates these approaches, allowing researchers to configure and access networks composed of emulated, simulated, and wide-area nodes and links. Netbed's overall design and implementation and demonstrates its ability to improve experimental automation and efficiency. Rich user interface, efficiency, and automation, enables qualitatively new kinds of experimentation across these mechanisms. It can be still improved.

2.4 Key-Policy Attribute Based Encryption

The primary drawback of the Sahai-Waters [3] threshold ABE system is that the threshold semantics are not very expressive and therefore are limiting for designing more general systems. In 2006, Goyal et al[4] introduced the idea of key-policy attribute-based encryption. In their construction a ciphertext is associated with a set of attributes and a user's key can be associated with any monotonic tree access structure. The construction of Goyal et al. can be viewed as an extension of the Sahai-Waters[3] techniques where instead of embedding a Shamir secret sharing scheme[13] in the private key, the authority embeds a more general secret sharing scheme for monotonic access trees.

2.5 Ciphertext–Policy ABE

In 2007, the first CP-ABE scheme was proposed by Bethencourt, Sahai and Waters [8]. They proposed a system for realizing complex access control on encrypted data that we call Cipher text-Policy Attribute-Based Encryption. They described an efficient system that was expressive in that it allowed an encryptor to express an access predicate f in terms of any monotonic formula over attributes. Here central authority generates the global key and issues the secret key for the user. Their approach has the drawbacks that it cannot guarantee security of data as server can be compromised.

In 2007, Ling Cheung [9], proposed CP-ABE schemes in which access structures are AND gates on positive and negative attributes. Here they introduced hierarchical attributes which reduced both cipher text size and encryption/decryption time while maintaining CPA security. Their approach has the drawbacks that it only allows a fixed number of system attributes and is limited to an AND gate (does not enable thresholds). These two limitations actually make it less expressive.

In 2011, Brent Waters[10] proposed a tool to prevent collusion attack, is to randomize each key with an freshly chosen exponent t. During decryption, each share will be multiplied by a factor t in the exponent. Intuitively, this factor “binds" the components of one user's key together so that they cannot be combined with another user's secret key components. In Brent Water [10], they use decryption key in the form of SK = (K = $g^{\alpha}h^{t}$,L=g^{t} ,K χ=$U^{t}\chi$,□χ ϵ S). However, the idea of using as the personalized information for the key owner to achieve traceability is infeasible.

However, in CP-ABE, the decryption privilege of a decryption key is shared by multiple users who possess the corresponding attributes, so that any malicious owner of a decryption key would have the intention or be very willing to leak partial or even his entire decryption privilege for financial interest or any other incentive, especially when there is no risk of getting caught is a issue of *Malicious Key Delegation*.

2.6 Access structure policy

In traditional access control schemes, a central authority can control a user's access to sensitive data. Firstly, since a user's identity needs to be validated by the authority, in a large distributed system, it is a difficult task to manage numerous users identities. Secondly, all users must trust the central authority. If the authority is malicious, he can impersonate any user without being detected. Being different from the traditional access control schemes, attribute-based access control [3], [8], [11], are the schemes that allow users to be validated by the descriptive attributes instead of their unique identities. Furthermore, a user can share his data by specifying an access structure so that all the users whose attributes satisfy it can access the data without knowing their identities. Therefore, attribute-based access control schemes are efficient primitives to share data with multiple users without knowing their identities.

Traditional encryption schemes cannot express a complex access policy, and additionally, the sender must know all the public keys of the receivers. Attribute-based encryption (ABE) introduced by Sahai and Waters [3] is a more efficient encryption scheme and it can express a complex access structure. Goyal, Pandey, Sahai and Waters proposed an ABE scheme[4] for fine-grained access policy where any monotonic access structure can be expressed by an access tree.

A monotonic access structure is an access structure where, given a universal set P, if a subset S□ of P satisfies the access structure, all subsets S of P which contain S□ satisfy the access structure. In an access tree, there is a tree access structure where interior nodes consist of AND and OR gates and the leaves consist of the attributes. Each interior node x of the tree specifies a threshold gate (kx, nx), where nx is the number of the children of x and kx □ nx.

Subsequently, Ostrovsky, Sahai and Waters proposed an ABE scheme [6] with a non-monotonic access structure where the secret keys are labeled with a set of attributes including not only the positive but also the negative attributes. Their access structure is complicated and less expressive. A (k, n)-threshold access structure is an access structure [3] where, given a universal set P with |P| = n, a subset S of P satisfies the access structure if and only if it contains at least k elements in P. This solves the collusion attack.

2.7 Decentralizing Attribute-Based Encryption

In 2011, Allison lewko [14],proposed a new multi-authority ABE scheme named decentralizing CP-ABE scheme. This scheme improved the previous multi-authority ABE schemes that require collaborations among multiple authorities to conduct the system setup. In this scheme, no cooperation between the multiple authorities is required in the setup stage and the key generation stage, and there is no

DOI: 10.18535/ijecs/v4i8.39

central authority. Note that the authority in this scheme can join or leave the system freely without reinitializing the system.

3. Designing Confidential Concurrent to Secure DBaaS

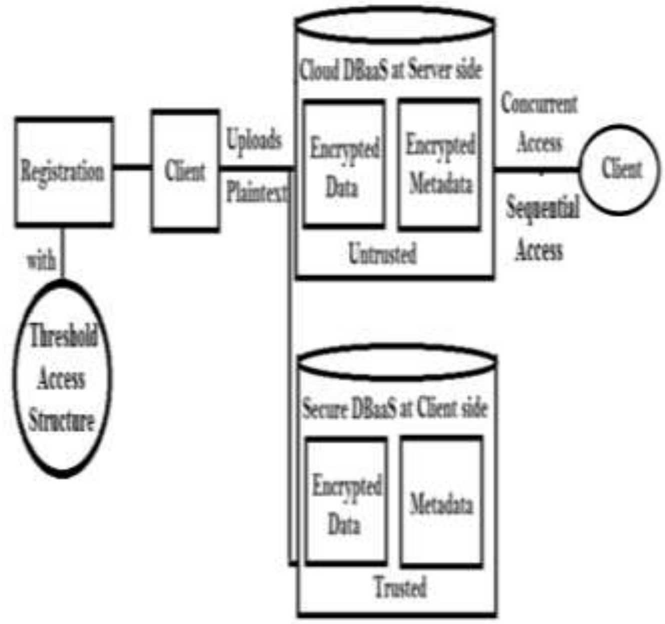


Figure 1: Confidential Concurrent to Secure DBaaS

3.1 Threshold Access structure

A (k, n)-threshold access structure [25] is an access structure where, given a universal set P with |P| = n, a subset S of P satisfies the access structure if and only if it contains at least k elements in P.

Operation of TAS

Here (k_x,n_x), $k_x <= n_x$

Where k is the number of children of x
Where n is the number of children of x
Where x is the interior nodes (AND,OR)

If $k_x = n_x$, AND gate and if $k_x = 1$ OR gate

If P={1,2,3,4} (elements of owner), |p|=n (number of elements in P); n=4

For Users S1={1,2,4} (N elements of owner-User Attribute Set)

For Access Policy S1'={1,2} (K elements of owner-associated with ciphertext for encryption &decryption)

One & only if S1 satisfies the S1' with range of K elements ciphertext will be decrypted.

Tolerance in corruption of authorities

At owner side,	**At user side,**
K=n-2=4-2=2	K=n-1=2-1=1
K=2	K=1

3.2 Secure Database as a Service

SecureDBaaS supports the execution of concurrent and independent operations to the remote encrypted database from many geographically distributed clients as in any unencrypted DBaaS setup. SecureDBaaS integrates existing cryptographic schemes, isolation mechanisms, and novel strategies for management of encrypted metadata on the untrusted cloud database. It allows cloud tenants to take full advantage of DBaaS qualities, such as availability, reliability, and elastic scalability, without exposing unencrypted data to the cloud provider**.** SecureDBaaS adopts multiple cryptographic techniques and isolation mechanism to transform plaintext data into encrypted tenant data and encrypted tenant data structures.

Pseudocode of S-DBaaS

- Plain name: the name of the corresponding column of the plaintext table.
- Coded name: the name of the column of the secure table. This is the only information that links a column to the corresponding plaintext column because column names of secure tables are randomly generated.
- Secure type: the secure type of the column. This allows a SecureDBaaS client to be informed about the data type and the encryption policies associated with a column.
- Encryption key: the key used to encrypt and decrypt all the data stored in the column.
- SecureDBaaS stores metadata in the metadata storage table that is located in the untrusted cloud as the database.
- Database and table metadata are encrypted through the same encryption key before being saved. This encryption key is called a master key.

3.3 Multi Authority ABE

Multi-authority ABE schemes we are aware of are Chase's original proposal [5] (which has already been discussed in Section B) and the very recent Lin *et al.* extension [12]. Lin, Cao, Liang and Shao proposed a multi-authority ABE scheme without a central authority [12] based on the distributed key generation (DKG) protocol and the joint zero secret sharing (JZSS) protocol [20]. To initialize the system, the multiple authorities must cooperatively execute the DKG protocol and the JZSS protocol twice and k times, respectively, where k is the degree of the polynomial selected by each authority. Each authority must maintain k + 2 secret keys. This scheme is K resilient, namely the scheme is secure if and only if the number of the colluding users is no more than k, and k must be fixed in the setup stage. Both schemes are KP-ABE and operate in a setting where multiple authorities are responsible for disjoint sets of attributes. The disadvantages of Chase's scheme have already been discussed in Section B.

The scheme of [12], like the scheme we will present here, has the advantage that it does not rely on a central authority. However, their scheme only achieves m-*resilience*, in that security is only guaranteed against a maximum of m colluding users. (In contrast, the results of [5] and our new results consider a much stronger model, which remains secure against any number of colluding users.) And this is not merely an issue of formal security: Lin *et al*. demonstrated a collusion attack of m+1 users[12]. In their scheme m is the number of secret keys that each authority obtains from a distributed key generation protocol. (This also means m must be determined when the system is initialized.) Clearly, for a large scale system, m should set reasonably high in order to guarantee security (a very loose desirable lower bound should be N2, where N is the number of authorities). This imposes burdens on the interactive distributed key generation protocol among all the authorities, and on their secure storage.

Pseudocode of PRKG

The generator is defined by the recurrence relation:

- X, where X is the sequence of pseudorandom values
- m, 0<m – the "modulus"
- a,0<a<m– the "multiplier“
- c,0<=c<m – the "increment"
- X_0,0<=X_0<m– the "seed" or "start value“
  are integer constants that specify the generator.

3.4 Management of Data

Encrypted tenant data are stored through secure tables into the cloud database. To allow transparent execution of SQL statements, each plaintext table is transformed into a secure table because the cloud database is untrusted.

3.5 Management of MetaData

Metadata generated by SecureDBaaS contain all the information that is necessary to manage SQL statements over the encrypted database in a way transparent to the user.

Two Types

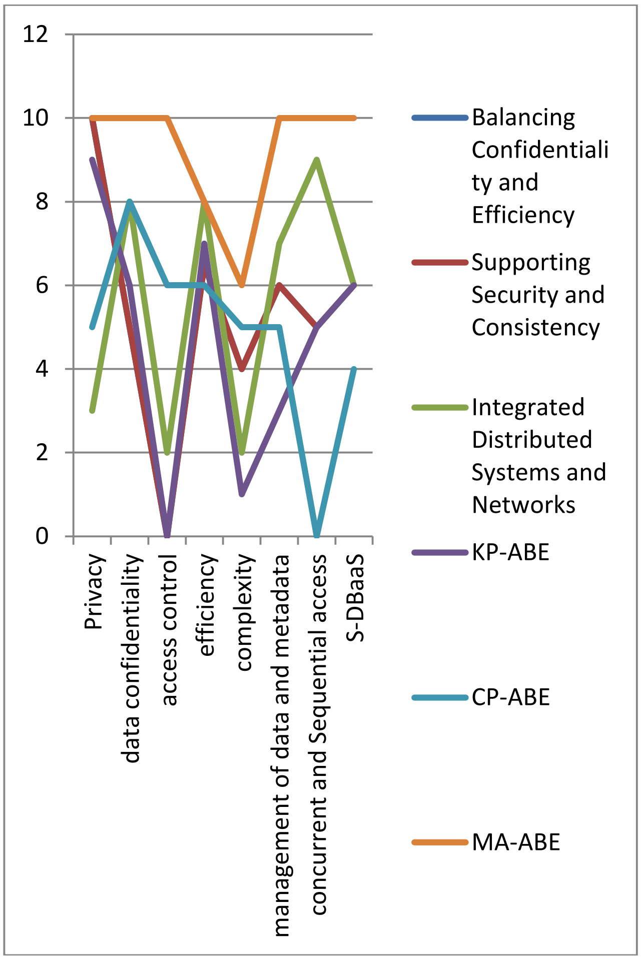
Database metadata are related to the whole database.

Table metadata contains all information that is necessary to encrypt and decrypt data of the associated secure table.

3.6 Concurrent and sequential SQL operations

It guarantees data confidentiality by allowing a cloud database server to execute concurrent SQL operations (not only read/write, but also modifications to the database structure) over encrypted data. It provides the same availability; elasticity and scalability of the original cloud DBaaS because it does not require any intermediate server. Multiple clients, possibly geographically distributed can access concurrently and independently a cloud database service. It does not require a trusted broker or a trusted proxy because tenant data and metadata stored by the cloud database are always encrypted.

4. Performance Analysis



5 CONCLUSION

An innovative architecture is proposed that guarantees the confidentiality of data stored in public cloud databases. It supports concurrent SQL operations on an attribute based encrypted data without compromisation as it is stored separately on both SDBaaS and Cloud Database so that any modification on cloud database cannot be altered in owner's SDBaaS. It also allows cloud tenants to take full advantage of DBaaS qualities, such as availability, reliability, without exposing unencrypted data to the cloud provider. It preserves data confidentiality and consistency at the client and cloud level; It also eliminates the intermediate server between the cloud client and the cloud provider.

REFERENCES

[1] M. Chase, “Multi-authority attribute based encryption,” in Proceedings: Theory of Cryptography Conference-TCC’07 (S. P. Vadhan,ed.), vol. 4392 of Lecture Notes in Computer Science, (Amsterdam, The Netherlands), pp. 515–534, Springer, Feb 21-24 2007.

[2] R. Ostrovsky, A. Sahai, and B. Waters, “Attribute- based encryption with non-monotonic access structures,” in Proceedings: ACM Conference on Computer and Communications Security-CCS’07 (P. Ning, S. D. C. di Vimercati, and P. F. Syverson)

[3] Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attribute- based encryption,” in Proceedings: IEEE Symposium on Security and Privacy (S & P’07), (Oakland, California, USA), pp. 321– 34, IEEE, May 20-23 2007.

[4] P. Mahajan, S. Setty, S. Lee, A. Clement, L. Alvisi, M. Dahlin, and M. Walfish, “Depot: Cloud Storage with Minimal Trust,” ACM Trans. Computer Systems, vol. 29, no. 4, article 12, 2011.

[5] R.A. Popa, C.M.S. Redfield, N. Zeldovich, and H. Balakrishnan, “CryptDB: Protecting Confidentiality with Encrypted Query Processing,” Proc. 23rd ACM Symp. Operating Systems Principles, Oct. 2011.

[6] H. Hacigu ¨mu ¨s¸,B.Iyer, C.Li, and S.Mehrotra, “Executing SQL over Encrypted Data in the Database-Service-Provider Model,” Proc. ACM SIGMOD Int’l Conf. Management Data, June 2002.

[7] J. Li and E. Omiecinski, “Efficiency and Security Trade-Off in Supporting Range Queries on Encrypted Databases,” Proc. 19th Ann. IFIP WG 11.3 Working Conf. Data and Applications Security, Aug. 2005.

[8] E. Mykletun and G. Tsudik, “Aggregation Queries in the Database-as-a-Service Model,” Proc. 20th Ann. IFIP WG 11.3 Working Conf. Data and Applications Security, July/Aug.2006.

[9] Lewko and B. Waters, “Decentralizing attribute - based encryption,” in Proceedings: Advances in Cryptology-EUROCRYPT’11 (K. G. Paterson, ed.), vol. 6632 of Lecture Notes in Computer Science, (Tallinn, Estonia), pp. 568–588, Springer, May 15-19 2011

[10] S. Muller, S. Katzenbeisser, and C. Eckert, “Distributed attribute based encryption,” in Proceedings: Information Security and Cryptology-ICISC’08 (P. J. Lee and J. H. Cheon, eds.), vol. 5461 of Lecture Notes in Computer Science, (Seoul, Korea), pp. 20–36, Springer, December 3-5 2008.

[11] Luca Ferretti, Michele Colanjanni, and Micre Marchetti, “Distributed, Concurrent, and Independent Access to Encrypted Cloud Databses” eds.), (Alexandria,Virginia, USA), pp. 195–203, ACM, October 28-31

[12] M. Armbrust et al., “A View of Cloud Computing,” Comm. of the ACM, vol. 53, no. 4, pp. 50-58, 2010.

[13] W. Jansen and T. Grance, “Guidelines on Security and Privacy in Public Cloud Computing,” Technical Report Special Publication 800-144, NIST, 2011.

[14] A.J. Feldman, W.P. Zeller, M.J. Freedman, and E.W. Felten, “SPORC: Group Collaboration Using Untrusted Cloud Re- sources,” Proc. Ninth USENIX Conf. Operating Systems Design and Implementation, Oct. 2010.

[15] J. Li, M. Krohn, D. Mazie `res, and D. Shasha, “Secure Untrusted Data Repository (SUNDR),” Proc. Sixth USENIX Conf.Opearting

[16] D. Agrawal, A.E. Abbadi, F. Emekci, and A. Metwally, “Database Management as a Service: Challenges and Opportunities,” Proc. 25th IEEE Int’l Conf. Data Eng., Mar.-Apr. 2009.

[17]V. Ganapathy, D. Thomas, T. Feder, H. Garcia-Molina, and R. Motwani, “Distributing Data for Secure Database Services,” Proc. Fourth ACM Int’l Workshop Privacy and Anonymity in the Information Soc., Mar. 2011.

[18]G. Cattaneo, L. Catuogno, A.D. Sorbo, and P. Persiano, “The Design and Implementation of a Transparent Cryptographic File System For Unix,” Proc. FREENIX Track: 2001 USENIX Ann. Technical Conf., Apr. 2001.

[19]E. Damiani, S.D.C. Vimercati, S. Jajodia, S. Paraboschi, and P. Samarati, “Balancing Confidentiality and Efficiency in Untrusted Relational Dbmss,” Proc. Tenth ACM Conf. Computer and Comm. Security, Oct. 2003.

[20]L. Ferretti, M. Colajanni, and M. Marchetti, “Supporting Security and Consistency for Cloud Database,” Proc. Fourth Int’l Symp. Cyberspace Safety and Security, Dec. 2012.

Author Profile

Blessy recieved B.E in Computer Science and Engineering from Scad College of Engineering and Technology, Anna University,2011, M.Tech Tech in Computer Science and Engineering from Hindustan University in 2013. She worked as an Assistant Professor in Department of Computer Science and Engineering, Scad Engineering College. She is currently Freelancing in Information Security related projects in Tirunelveli, Tamilnadu, India.

Varun Chand received the B.Tech in Information Technology from Mahatma Gandhi University in 2007, M.Tech in Computer Science and Engineering from Karunya University in 2010 and MBA in Human Resource from Mahatma Gandhi University in 2013. He now works as Assistant Professor in Department of Information Technology, College of Engineering and Management, Punnapra, Kerala, India.

DOI: 10.18535/ijecs/v4i8.39