International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 07
June 2017

Revocable-Storage Identity-Based Encryption in Cloud Computing

1. A. KALPANA 2. B.N.V. MADHUBABU 3. DR. K NAGESWARARAO

[1]Pg Scholar, Department of CSE, Mother Teresa Institute of Science and Technology, Sathupally
kalpana.avuluri@gmail.com.

[2]Associate Professor , Department of CSE, Mother Teresa Institute of Science and Technology, Sathupally
bnvmadhubabu2014@gmail.com.

[3] Professor & HOD, Department of CSE, Mother Teresa Institute of Science and Technology, Sathupally
nageswararaokapu@yahoo.com.

ABSTRACT— Cloud computing provides a flexible and convenient way for data sharing, which brings various benefits for both the society and individuals. But there exists a natural resistance for users to directly outsource the shared data to the cloud server since the data often contain valuable information. Thus, it is necessary to place cryptographically enhanced access control on the shared data. Identity-based encryption is a promising cryptographical primitive to build a practical data sharing system. However, access control is not static. That is, when some user's authorization is expired, there should be a mechanism that can remove him/her from the system.

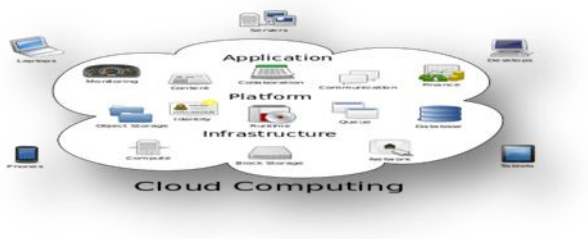
Consequently, the revoked user cannot access both the previously and subsequently shared data. To this end, we propose a notion called revocable-storage identity-based encryption (RS-IBE), which can provide the forward/backward security of ciphertext by introducing the functionalities of user revocation and ciphertext update simultaneously. Furthermore, we present a concrete construction of RS-IBE, and prove its security in the defined security model. The performance comparisons indicate that the proposed RS-IBE scheme has advantages in terms of functionality and efficiency, and thus is feasible for a practical and cost-effective data-sharing system. Finally, we provide implementation results of the proposed scheme to demonstrate its practicability.

1 INTRODUCTION

What is cloud computing?

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and

computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services.



Structure of cloud computing

How Cloud Computing Works?

The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive computer games.

The cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing.

Characteristics and Services Models:

The salient characteristics of cloud computing based on the definitions provided by the National Institute of Standards and Terminology (NIST) are outlined below:

- **On-demand self-service**: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.
- **Broad network access**: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).
- **Resource pooling**: The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned

according to consumer demand. There is a sense of location-independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or data center). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.

- **Rapid elasticity**: Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.
- **Measured service**: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be managed, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

Services Models:

Cloud Computing comprises three different service models, namely Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). The three service models or layer are completed by an end user layer that encapsulates the end user perspective on cloud services. The model is shown in figure below. If a cloud user accesses services on the infrastructure layer, for instance, she can run her own applications on the resources of a cloud infrastructure and remain responsible for the support, maintenance, and security of these applications herself. If she accesses a service on the application layer, these tasks are normally taken care of by the cloud service provider.

Benefits of cloud computing:

1. **Achieve economies of scale** – increase volume output or productivity with fewer people. Your cost per unit, project or product plummets.

2. **Reduce spending on technology infrastructure.** Maintain easy access to your information with minimal upfront spending. Pay as you go (weekly, quarterly or yearly), based on demand.
3. **Globalize your workforce on the cheap.** People worldwide can access the cloud, provided they have an Internet connection.
4. **Streamline processes.** Get more work done in less time with less people.
5. **Reduce capital costs.** There's no need to spend big money on hardware, software or licensing fees.
6. **Improve accessibility.** You have access anytime, anywhere, making your life so much easier!
7. **Monitor projects more effectively.** Stay within budget and ahead of completion cycle times.
8. **Less personnel training is needed.** It takes fewer people to do more work on a cloud, with a minimal learning curve on hardware and software issues.
9. **Minimize licensing new software.** Stretch and grow without the need to buy expensive software licenses or programs.
10. **Improve flexibility.** You can change direction without serious "people" or "financial" issues at stake.

Advantages:

1. **Price:** Pay for only the resources used.
2. **Security**: Cloud instances are isolated in the network from other instances for improved security.
3. **Performance:** Instances can be added instantly for improved performance. Clients have access to the total resources of the Cloud's core hardware.
4. **Scalability:** Auto-deploy cloud instances when needed.
5. **Uptime:** Uses multiple servers for maximum redundancies. In case of server failure, instances can be automatically created on another server.
6. **Control:** Able to login from any location. Server snapshot and a software library lets you deploy custom instances.

7. **Traffic:** Deals with spike in traffic with quick deployment of additional instances to handle the load.

2 Related work

2.1 Revocable identity-based encryption

The concept of identity-based encryption was introduced by Shamir and conveniently instantiated by Boneh and Franklin . IBE eliminates the need for providing a public key infrastructure (PKI). Regardless of the setting of IBE or PKI, there must be an approach to revoke users from the system when necessary, e.g., the authority of some user is expired or the secret key of some user is disclosed. In the traditional PKI setting, the problem of revocation has been well studied and several techniques are widely approved, such as certificate revocation list or appending validity periods to certificates. However, there are only a few studies on revocation in the setting of IBE. Boneh and Franklin first proposed a natural revocation way for IBE. They appended the current time period to the ciphertext, and non-revoked users periodically received private keys for each time period from the key authority. Unfortunately, such a solution is not scalable, since it requires the key authority to perform linear work in the number of non-revoked users. In addition, a secure channel is essential for the key authority and non-revoked users to transmit new keys. To conquer this problem, Boldyreva, Goyal and Kumar introduced a novel approach to achieve efficient revocation. They used a binary tree to manage identity such that their RIBE scheme reduces the complexity of key revocation to logarithmic (instead of linear) in the maximum number of system users. However, this scheme only achieves selective security. Subsequently, by using the aforementioned revocation technique, Libert and Vergnaud [proposed an adaptively secure RIBE scheme based on a variant ofWater's IBE scheme, Chen et al. constructed a RIBE scheme from lattices. Recently, Seo and Emura proposed an efficient RIBE

scheme resistant to a realistic threat called decryption key exposure, whichmeans that the disclosure of decryption key for current time period has no effect on the security of decryption keys for other time periods. Inspired by the above work and , Liang et al. introduced a cloud-based revocable identity-based proxy re-encryption that supports user revocation and ciphertext update. To reduce the complexity of revocation, they utilized a broadcast encryption scheme [2to encrypt

the ciphertext of the update key, which is independent of users, such that only non-revoked users can decrypt the update key. However, this kind of revocation method cannot resist the collusion of revoked users and malicious non-revoked users as malicious non revoked users can share the update key with those revoked users. Furthermore, to update the ciphertext, the key authority in their scheme needs to maintain a table for each user to produce the re-encryption key for each time period, which significantly increases the key authority's workload.

3. PRELIMINARIES:

3.1. DECISIONAL BDHE ASSUMPTION:

The decisional ℓ-BDHE problem is formalized as follows. Choose a group G 1with prime order p according to the security parameter λ select a generator g of G1 and a, s<- R ZP and let f i=g a i. .T Provide the vector f= (g, g s , f 1, ..., f ℓ , f ℓ+2 , ..., f 2ℓ) and an element D ∈ G 2 to a probabilistic polynomial time (PPT) algorithm C, it outputs 0 to indicate that D = e(g s , g aℓ+1) , and outputs 1 to indicate that D is a random element from G 2.

3.2. KUNODES ALGORITHM:

By using this algorithm only on revoked user at a time period are able to decrypt the cipher text

INPUT: Binary tree revocation list,Time period

OUTPUT : outputs the smallest subset Y of nodes of BT such that Y contains an ancestor for each node that is not revoked before the time period

STEPS:

1. Data owner upload the file in cloud with validity time

2. Data user access the data.

    2.1. if the user tries to access the data within a specified time only he is able to access the data

    2.2. Otherwise data owner need to update the key.

3.Data owner update the key used by the user.

4. Then he will update the cipher text. This will provide both forward and backward security to the data stored in a cloud. DOI: 10.18535/ijecs/v6i3.29 Sonia Jenifer Rayen

WORKING:

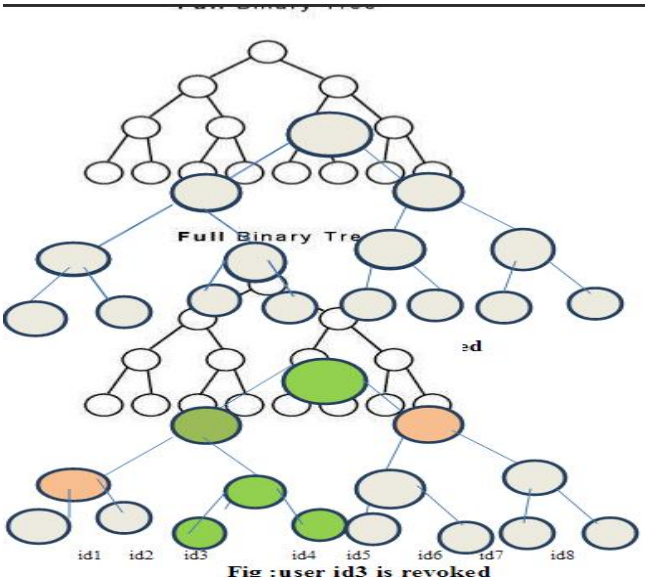


Fig :user id3 is revoked

By this algorithm ,when we rovoke the leaf node(id3) their ancestors also get updated(nod es in green color) and the node which shares the same key of revoked nod(nodes in orange color)e also get updated.

Algorithm 1

KUNodes(BT , RL, t)

1:X,Y←−∅

2:for all (ηi,ti)∈ RL

do

3: if ti≤t

then

4:Add Path(η i) to X

5: end if

6: end for

7: for all θ ∈ X do

8: if θ l ∈ /X then

9: Add θ l to Y

10: end if

11: if θ r ∈ /X then

12: Add θ r to Y

13: end if

14: end for

15: if Y= ∅ then

16: Add the root node ε to Y

17: end if

18: return Y

IV CONCLUSIONS

Cloud computing brings great convenience for people. Particularly, it perfectly matches the increased need of sharing data over the Internet. In this paper, to build a cost-effective and secure data sharing system in cloud computing, we proposed a notion called RS-IBE, which supports identity revocation and ciphertext update simultaneously such that a revoked user is prevented from accessing previously shared data, as well as subsequently shared data. Furthermore, a concrete construction of RS-IBE is presented. The proposed RS-IBE scheme is proved adaptive-secure in the standard model, under the decisional ℓ-DBHE assumption. The comparison results demonstrate that our scheme has advantages in terms of efficiency and functionality, and thus is more feasible for practical applications.

REFERENCES

[1] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 50–55, 2008.

[2] iCloud. (2014) Apple storage service. [Online]. Available: https://www.icloud.com/

[3] Azure. (2014) Azure storage service. [Online]. Available: http://www.windowsazure.com/

[4] Amazon. (2014) Amazon simple storage service (amazon s3). [Online]. Available: http://aws.amazon.com/s3/

[5] K. Chard, K. Bubendorfer, S. Caton, and O. F. Rana, "Social cloud computing: A vision for socially motivated resource sharing," *Services Computing, IEEE Transactions on*, vol. 5, no. 4, pp. 551–563, 2012.

[6] C. Wang, S. S. Chow, Q. Wang, K. Ren, and W. Lou, "Privacypreserving public auditing for secure cloud storage," *Computers, IEEE Transactions on*, vol. 62, no. 2, pp. 362–375, 2013.

[7] G. Anthes, "Security in the cloud," *Communications of the ACM*, vol. 53, no. 11, pp. 16–18, 2010.

[8] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 24, no. 9, pp. 1717–1726, 2013.

[9] B. Wang, B. Li, and H. Li, "Public auditing for shared data with efficient user revocation in the cloud," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 2904–2912.

[10] S. Ruj, M. Stojmenovic, and A. Nayak, "Decentralized access control with anonymous authentication of data stored in clouds," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 2, pp. 384–394, 2014.

[11] X. Huang, J. Liu, S. Tang, Y. Xiang, K. Liang, L. Xu, and J. Zhou, "Cost-effective authentic and anonymous data sharing with forward security," *Computers, IEEE Transactions on*, 2014, doi: 10.1109/TC.2014.2315619.

[12] C.-K. Chu, S. S. Chow, W.-G. Tzeng, J. Zhou, and R. H. Deng, "Key-aggregate cryptosystem for scalable data sharing in cloud storage," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 2, pp. 468–477, 2014.

[13] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in cryptology*. Springer, 1985, pp. 47–53.

[14] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586– 615, 2003.

[15] S. Micali, "Efficient certificate revocation," Tech. Rep., 1996.

[16] W. Aiello, S. Lodha, and R. Ostrovsky, "Fast digital identity revocation," in *Advances in Cryptology–CRYPTO 1998*. Springer, 1998, pp. 137–152.

[17] D. Naor, M. Naor, and J. Lotspiech, "Revocation and tracingschemes for stateless receivers," in *Advances in Cryptology– CRYPTO 2001*. Springer, 2001, pp. 41–62.