ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 7, Special Issue 6, May 2018

2nd National Conference on "RECENT TRENDS IN COMPUTER SCIENCE & INFORMATION TECHNOLOGY (RTCSIT'18)"

13th-14th March 2018

Organized by

Departments of Computer Science & Information Science Engineering, Sri Krishna Institute of Technology, Bangalore, India

Cloud++: A Secure and Timed Data Access Control Scheme for Cloud

Rudra Gowda M Patil[1], Dhananjaya V[2]

Student, Department of Computer Science and Engineering, Sri Krishna Institute of Technology, Bengaluru, India[1]

Professor, Department of Computer Science and Engineering, Sri Krishna Institute of Technology, Bengaluru, India[2]

ABSTRACT:Secure cloud storage, which is a rising cloud service, is intended to ensure the secrecy of outsourced information yetadditionally to give adaptable information access to cloud clients whose information is out of physical control. Ciphertext-Policy Attribute-Based Encryption(CP-ABE) is viewed as a standout amongst the most encouraging systems that might be utilized to secure the assurance of the service. Be that as it may,the utilization of CP-ABE may yield an unavoidable security rupture which is known as the misuse of access credential (i.e. decryption right). In this paper, we examine the two primary instances of access credential abuse: one is on the semi-trusted authority side, and the other is in favour of cloud client. To relieve the misuse, we propose revocable CP-ABE based cloud storage framework with explicit revoking, timed data accessing and multiple auditing abilityreferred asCloud++.

KEYWORDS:CP-ABE, Access Credentials Misuse, Traceability andRevocation, Auditing, TimeEncoding.

I. INTRODUCTION

The predominance of cloud computing may in a roundabout way cause of the helplessness to the privacy of outsourced data furthermore, the protection of cloud clients. A specific test here is on the most proficient method to ensure that exclusive approved clients can pick up access to the data, which has been outsourced to cloud, at anyplace and whenever. One credulous arrangement is to utilize encryption system on the data before uploading to cloud. Yet it limits data sharing scheme.Thisis so in light of the fact that a data proprietor needs to download the encrypted data from cloud and further re-scramble them for sharing (assume the data proprietor has no nearby duplicates of the data). A fine-grained get to control over encrypted data is alluring with regards to cloud computing.

Ciphertext-Policy Attribute-Based Encryption (CP-ABE) might be an effective answer for ensuring the confidentiality of data and to give fine-grained access control here. In a CP-ABE based cloud storage framework, for instance, associations and people can first specify access policy over attributes of a potential cloud client. Approved cloud clients at that point are conceded access credentials (i.e., decryption keys) corresponding to their attributesets which can be utilized to get access to the outsourced data. CP-ABE offers a dependable strategy to protect data put away in cloud, yet likewise empowers fine-grained access control over the data.

Any information that is stored in cloud if leaked, could result in arange of consequences for the association and people. The existing CP-ABE based[1] schemeenables us to keep security breach from outside attacker and also an insider of the association who commits the "crimes" of redistributingthe decryption rights and the circulation of understudy data in plain arrangement for illicit financial picks ups. At the same time, it can also ensure that semi-trusted authority won't (re-)distribute the created access credentials to others by proposing CryptCloud+, which provided an accountable authority and revocable CP-ABE based cloud storage system. In any case, one trying issue in taking care of client disavowal in cloud storage is that a revoked client may in any case will still have the capacity to unscramble an old ciphertext they were approved to access before being revoked. To address this issue, the ciphertext put away in the cloud storage ought to be updated, preferably by the (untrusted) cloud server. Also it lacked timed data accessing control which would provide a substantial level of security.

This paper proposes Cloud++, an extended model of the CryptCloud+by providing a timed data access control. And multiple auditing and comparison scheme and removed one of the two revocable systems and reduced it to one explicit revocable system. We have incorporated Shengmin's et. al [2] time encoding mechanism into the revocable system to provide timed data access control.This paper extends work in [1] as follows.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 7, Special Issue 6, May 2018

2nd National Conference on "RECENT TRENDS IN COMPUTER SCIENCE & INFORMATION TECHNOLOGY (RTCSIT'18)"

13th-14th March 2018

Organized by

Departments of Computer Science & Information Science Engineering, Sri Krishna Institute of Technology, Bangalore, India

1) We address a shortcoming in the auditing methodology in [1] . In particular, the auditing methodology will flop in this case just auditor said to be trusted completely may neglect to be straightforward on occasion. As a moderation, we modify the auditing method and add different auditors rundown to review and think about outcomes.
1) We improve the usefulness of the development ATER-CP-ABE in [1]. This ATR-CP-ABE development enable us to adequately revoke the noxious clients expressly.
Paper is organized as follows. Section II describes all the related work done previously. Description of Architectural model of Cloud++: A Secure and Timed Data Access Control Scheme for Cloud is given in Section III. Section IV presents experimental results. Finally, Section V presents conclusion.

II. RELATED WORK

In the paper [1] creators explored the two primary instances of access qualification abuse: one is on the semi-trusted expert side, and the other is in favor of cloud client. To moderate the abuse, authors proposed the primary responsible specialist and revocable CP-ABE based distributed storage framework with white-box traceability and reviewing, alluded to as CryptCloud+.
In the paper [2], authors outlined a proficient revocable attribute based encryption (RABE) plot with the property of ciphertext designation.propose a protected and effective fine-grained get to control and information sharing plan for dynamic client bunches by (1) characterizing and upholding access approaches in view of the characteristics of the information; (2) allowing key generation center (KGC) to effectively refresh client certifications for dynamic client gatherings; and (3) permitting some costly calculation errands to be performed by untrusted CSPs without requiring any delegation key.
In the paper[3],creator proposed the Secure Information Sharing in Clouds (SeDaSC) methodology that givesinformation privacy and honesty, get to controlinformation sharing (sending) without utilizing figure concentrated re-encryption, insider risk securityandforward and in reverse access control.
This paper [4] built up another cryptosystem for fine-grained sharing of scrambled information. In this cryptosystem, ciphertexts are named with sets of characteristics and private keys are related with get to structures that control which ciphertexts a client is capable to decode.
This paper [5], proposed a multi-expert ciphertext-approach ABE conspire with responsibility, which permits following the personality of a getting into mischief client who released the decoding key to others, and in this manner diminishes the confide in presumptions on the specialists as well as the clients.
This paper [6], proposes an idea called auditable σ-time outsourced CP-ABE, which is accepted to be appropriate to distributed computing. In this a costly blending activity caused by decoding is offloaded to cloud and in the interim, the rightness of the task can be inspected effectively.
This paper [7], provided an expressive, productive and revocable information get to control plot for multi-expert distributed storage frameworks, where there are various authorities exist together and every specialist can issue properties autonomously. In particular, they proposed a revocable multi-expert CP-ABE conspire, and applied it as the fundamental systems to outline the data access to control plot.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 7, Special Issue 6, May 2018

2nd National Conference on "RECENT TRENDS IN COMPUTER SCIENCE & INFORMATION TECHNOLOGY (RTCSIT'18)"

13th-14th March 2018

Organized by

Departments of Computer Science & Information Science Engineering, Sri Krishna Institute of Technology, Bangalore, India

III. METHODOLOGY

Our approach of finding malicious user and preventing revoked user from further accessing the ciphertext is shown below.

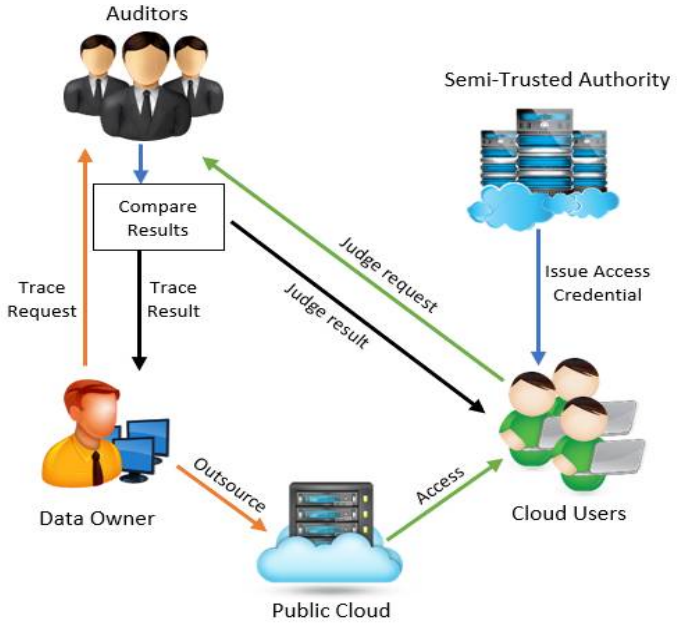

Fig.3.1 Architecture of Cloud++ System.

Firstly, we shall describe the architecture of new Cloud++ data access control system. Fig.3.1 above shows the architecture of our system. System contains five modules which are described below.

1. Data Owner:Data Owners (DOs) encrypts their information under the significant access policies before outsourcing the (encrypted) information to an open cloud (Public Cloud).
2. Public Cloud Storage: An untrusted cloud storage service provider where data outsourced by DO will be stored.
3. Cloud Users: CU's are end-users who requests for access to outsourced files.
4. Auditors: They are the Third-Party auditors who verifies, audits, traces cloud users and revokes malicious users of the cloud system.
5. Semi-Trusted Authority: They are responsible for setting up the system, generation of keys and redistribution of keys to authorized and unauthorized users.

Semi-Trusted Authority is semi-trusted as in it might (re-)convey access credentials to the individuals who are unapproved however creates framework parameters (to be imparted to Auditor) sincerely. A completely trusted Auditor keeps a duplicate of the framework parameters shared by Semi-Trusted Authority. Data Owners encode their information to avert unapproved get to. Approved DUs may deliberately release their entrance certifications,for example, pitching accreditations to an outsider. Practically speaking, get to accreditations are probably going to pull in potential purchasers (in dark showcase), and the framework deceivers (offering the certifications) may never have been gotten.

Even though auditors claim to be fully trusted they are never the case. At least one in three shall be a dishonest auditor who wishfully provide incorrect audit results to the data owners. In order to mitigate this, we propose a multiple auditor scheme whose independent audits are compared and the correct audit results are sent to the authorized data owners. Request of tracing from data owners is received by all the auditors and auditing is performed from every independent auditor. These audit results are compared for maximum accuracy in results. Suppose one found to be an improper audit result, then such an auditor is revoked from the system. And the audit result with proper audit is sent back to the data owner.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 7, Special Issue 6, May 2018

2nd National Conference on "RECENT TRENDS IN COMPUTER SCIENCE & INFORMATION TECHNOLOGY (RTCSIT'18)"

13th-14th March 2018

Organized by

Departments of Computer Science & Information Science Engineering, Sri Krishna Institute of Technology, Bangalore, India

We also revise the ATER-CP-ABE algorithm provided in paper[1] by incorporating time encoding algorithm provided by Shengmin et,al [2] in his paper.
Our ATR-CP-ABE has following algorithms.

- **Setup**(*Lamda, U*) = (*Pp, msk*): Security parameter P_S and universe of attributes U is given as input which outputs public parameters Pp and a Master secret key msk. Also it initializes an empty revocation list RL.
- **TimeEncode**(*Date, T*) = t_{en}: Given the input *T*(bounded system life), *Date* (current date when key is generated) we get bit string $t_{en}$of the size $\log_2 T$ as output.
- **TimeDecode**($sk_{id,S,+ten}$,*cDate, cTime*) = $sk_{id,S,-ten}$.Given the input *cDate*(current date during decryption), *cTime*(current time during decryption), we get secret key $sk_{id,S,-ten}$ without time component.
- **KeyGen**(*Pp, msk, id, S,* t_{en}) = $sk_{id,S,+ten}$: Key generation phase between Semi trusted Authority and Users. We give public parameter *Pp*, master secret key *msk* and set of attributes *S* with use identity *id* along with encoded time bits t_{en}. We get a Secret key$sk_{id,S,+ten}$ as output which is associated with user id.
- **Encrypt**(*Pp, msg, AS, RL*) = *cipherText* : On input of *Pp*, a plaintext message *msg*, an access structure *AS* over the universe of attributes, and a revocation list *RL*, it outputs aciphertext *cipherText*.
- **Decrypt**(*Pp,* $sk_{id,S,+ten}$*, cipherText, cDate, cTime*) = *msg* | *error* : On input of public parameter *Pp* , asecret key $sk_{id,S,+ten}$, and a ciphertext *cipherText*, it first extracts time component of the secret key by performing **TimeDecode**($sk_{id,S,+ten}$,*cDate, cTime*) and outputs a secret key $sk_{id,S,-ten}$.Then checks if the decoded time component is still less than current *cDate* and *cTime*.If suppose the time component is less than current date and time during decryption, then algorithm proceeds to find plaintext. It outputs theplaintext *msg* if the attribute set S of $sk_{-ten,id}$ satisfies theaccess structure of *cipherText* and *id* does not belong to*RL*. Otherwise, itoutputs *error*.
- **KeyFormCheck**(*Pp, sk*) = 1 | 0 : 1=wellFormed 0= weaklyFormed. On input *Pp* and a secret key $sk_{id,S,+ten}$, it outputs 1 if $sk_{id,S,+ten}$ passes the key sanity check. Otherwise, it outputs 0.
- **Trace**(*Pp, msk, sk*) = sk^*_{id} | noTraceReq : On input *Pp, msk* and a secret key *sk*, it first checks whether *sk* is well-formed in order to further determine whether *sk* needs to be traced. A secret key sk defined as wellformed if KeyFormCheck(*Pp, sk*)= 1. For a wellformed *sk*, it extracts the identity from *sk*. It thenoutputs an identity with which the *sk* associates, and places it in the revocation list *RL*. Otherwise, itoutputs a noTraceReq symbol indicating that *sk* does not needto be traced.
- **Audit**(*Pp,* $sk_{id,S,+ten}$*,* sk^*_{id}) = 1 | 0 1=guilty 0=innocent. If found guilty, user is revoked and decryption key is regenerated by performing KeyGen(*Pp, msk, id, S,* t_{en}) with new time encoded parameter.

Our Cloud++ works as follows:

- **System setup**: Semi-Trusted Authority setups the system. It runs **Setup**(*Lamda, U*) = (*Pp, msk*)to generate system publicparameter *Pp* and master secret key *msk*. It shares this *Pp* and *msk* with Auditors before publishing *Pp* to others.
- **User registration**: A user gets successfully registered to the system and is issued an identity *id* an attribute set S is assigned to user. Semi-Trusted Authority generates first generates an encoded time bits which specifies the expiration of access to the data by calling TimeEncode(*Date, T*) = $t_{en.}$ He then generates a secret access credential for the user based on his identity *id* and attribute set *S* and combines time encoded bit strings to the encrypted access key by calling KeyGen(*Pp, msk, id, S,* t_{en}) = $sk_{id,S,+ten}$.
- **FileOutsource**: Data Owner first encrypts his file using AES data encryption technique and generates a session key m_{skey} and defines access policy and encrypts the data by calling **Encrypt**(*Pp,* m_{skey}*, AS, RL*) = *cipherText*.
- **File Access**: Data user requests for the file access. Cloud returns requested file to data user. Data user calls **Decrypt**(*Pp,* $sk_{id,S,+ten}$*, cipherText, cDate, cTime*) = *msg* | *error* to decrypt the encrypted data.
- **Trace**: When auditor finds a secret access credential is being sold online or receives a trace request from a DataOwner, it runs **Trace**(*Pp, msk, sk*) = sk^*_{id} | noTraceReq to find out who the leaker is.
- **Audit**: When a data user with identity id is traced as the leaker but claims innocence, it sends an audit request along with his/her access credential to auditors. Upon receiving the audit request, Auditors calls guilty or innocent **Audit**(*Pp,* $sk_{id,S,+ten}$*,* sk^*_{id}) = 1 | 0 to determine whether the (accused) user is indeed innocent, where access credentialsk^*_{id} is the leaked access credential.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 7, Special Issue 6, May 2018

2nd National Conference on "RECENT TRENDS IN COMPUTER SCIENCE & INFORMATION TECHNOLOGY (RTCSIT'18)"

13th-14th March 2018

Organized by

Departments of Computer Science & Information Science Engineering, Sri Krishna Institute of Technology, Bangalore, India

IV. EXPERIMENTAL RESULTS

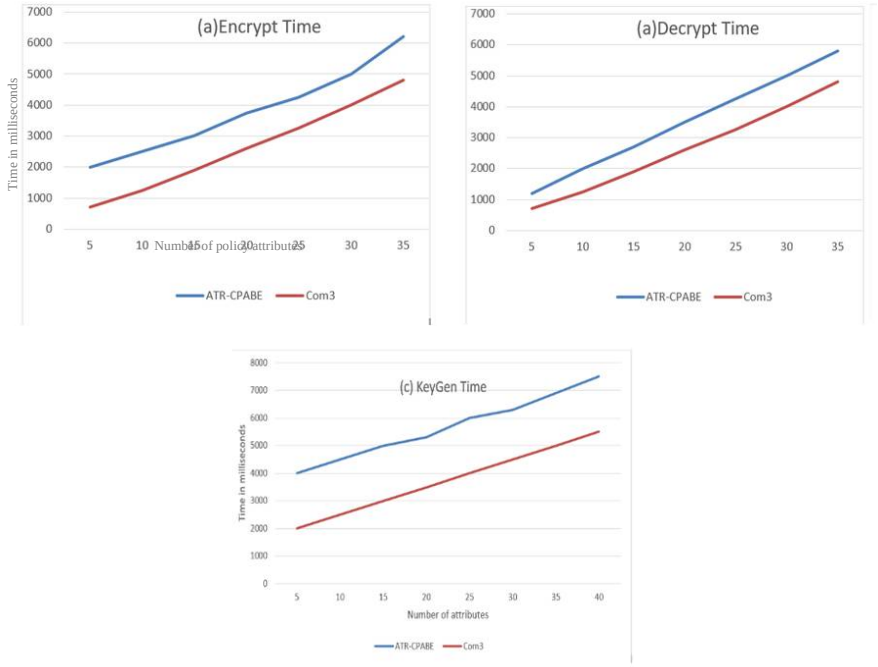

Fig.4.1 Experimental Results

In CP-ABE frameworks, the multifaceted nature of ciphertext arrangement impacts both the encryption time and the unscrambling time. To represent this, we produce ciphertext approaches in the type of (S1 and S2 ... what's more, Sl) to re-enact the most pessimistic scenario circumstance, where Si is a trait. We endeavor to assess the effectiveness of ATR-CP-ABE looking at the aggregate time taken amid each phase with the unique CP-ABE conspire which does not consider the entrance accreditations mishandle issue and the disavowal issue. As delineated in Fig. 4.1, we look at the time cost of executing singular stage (counting the Encrypt Time, the Decrypt Time and the KeyGen Time (of AT)). Since we consider both access qualification mishandle issue and the denial issue, it isn't shocking to watch that our frameworks require additional time.

V. CONCLUSION

In this work, we have tried to solve the problem of credential leakage in CP-ABE based distributed storage framework by planning a accountable authority and revocable Cloud++. This is the CP-ABE based distributed storage framework that supports accountable authority, various inspecting and powerful revocation. Specifically, Cloud++ enables us to follow and disavow noxious cloud clients. Our approach can be likewise utilized as a part of the situation where the clients' certifications are redistributed by the semi-put stock in specialist. This likewise gives a timed data access control where client can get to the information inside a determined time for the given key in this manner preventing access to documents by the renounced user. Furthermore, AU is thought to be completely confided in CryptCloud+. Be that as it may, practically speaking, it may not be the situation. so we gave an approach to decrease trust from AU by utilizing various AU's.

REFERENCES

[1] "CryptCloud+: Secure and Expressive Data Access Control for Cloud Storage" Jianting Ning, Zhenfu Cao, Senior Member, IEEE, Xiaolei Dong, Kaitai Liang, Member, IEEE, Lifei Wei, and Kim-Kwang Raymond Choo, Senior Member, IEEE

[2] "Secure Fine-Grained Access Control and Data Sharing for Dynamic Groups in Cloud" Shengmin Xu, Guomin Yang , Senior Member, IEEE, Yi Mu, Senior Member, IEEE,and Robert H. Deng Fellow, IEEE

[3] Mazhar Ali, Revathi Dhamotharan, Eraj Khan, Samee U. Khan, Athanasios V. Vasilakos, Keqin Li, and Albert Y. Zomaya. "Sedasc: Secure data sharing in clouds." IEEE Systems Journal, 11(2):395–404,2017.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization *Volume 7, Special Issue 6, May 2018*

2nd National Conference on "RECENT TRENDS IN COMPUTER SCIENCE & INFORMATION TECHNOLOGY (RTCSIT'18)"

13th-14th March 2018

Organized by

Departments of Computer Science & Information Science Engineering, Sri Krishna Institute of Technology, Bangalore, India

[4] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. "Attribute-based encryption for fine-grained access control of encrypted data." In Proceedings of the 13th ACM conference on Computerand communications security, pages 89–98. ACM, 2006.

[5] Jin Li, Qiong Huang, Xiaofeng Chen, Sherman SM Chow, Duncan S Wong, and Dongqing Xie. "Multi-authority ciphertext-policy attribute-based encryption with accountability." In Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, ASIACCS 2011, pages 386–390. ACM, 2011.

[6] Jianting Ning, Zhenfu Cao, Xiaolei Dong, Kaitai Liang, Hui Ma, and LifeiWei. 'Auditable -time outsourced attribute-based encryption for access control in cloud computing." IEEE Transactions on Information Forensics and Security, 13(1):94–105, 2018.

[7] Kan Yang, Student Member, IEEE, and Xiaohua Jia, Fellow, IEEE. "Expressive, Efficient, and Revocable Data Access Control for Multi-Authority Cloud Storage."

[8] Amit Sahai and Brent Waters. "Fuzzy identity-based encryption." In Advances in Cryptology–EUROCRYPT 2005, pages 457–473.Springer, 2005.

[9] Brent Waters. "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization." In Public Key Cryptography–PKC 2011, pages 53–70. Springer, 2011.

[10] Kan Yang, Zhen Liu, Xiaohua Jia, and Xuemin Sherman Shen. "Time-domain attribute-based access control for cloud-based video content sharing: A cryptographic approach." IEEE Transactions on Multimedia, 18(5):940–950, 2016.

[11] Zhen Liu, Zhenfu Cao, and Duncan S Wong. "Blackbox traceable cp-abe: how to catch people leaking their keys by selling decryption devices on ebay." In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, pages 475–486. ACM, 2013.