IJDCST @Aug-Sept-2015, Issue- V-3, I-6, SW-05
ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

Allowing data integrity along with security in clouds

B.Venkata Subba Reddy[1], T P V V Srinivasa Rao[2], Dr.J.Srinivasa Rao[3]

[1] M.Tech (CSE), Nova College of Engineering & Technology, A.P., India.

[2] Assistant Professor , Dept. of Computer Science & Engineering, Nova College of Engineering & Technology, A.P., India.

[3]Professor, Dept. of Computer Science & Engineering, Nova College of Engineering & Technology, A.P., India.

Abstract: In cloud computing, data is moved to a remotely located cloud server. Clouds provide an inexpensive access to remote resources. Cloud faithfully stores the data and return back to the owner whenever needed. Cloud storage moves application software and databases to the centralized to large data centers on which user does not have any control, where the management of data and services are not fully trustable. However, this unique feature of the cloud poses many new security challenges on storing the data remotely without having any backup. In this paper, open challenging issue like data security and data integrity are addressed and resolved by allowing the data owner to delegate most of the computation tasks involved in data access control to untrusted cloud servers without disclosing the underlying data contents. We achieve the goal by exploiting and uniquely combining techniques encryption. Proposed system focus on multi keyword ranked search over encrypted in cloud data with maximum number of results.

Keywords: Cloud computing, tanking search, data integrity.

I INTRODUCTION

Internet has been a driving force towards the various technologies that have been developed since its inception. Arguably, one of the most discussed among all of them is *Cloud Computing*. Over the last few years, cloud computing paradigm has witnessed an enormous shift towards its adoption and become a trend in the information technology market as it promises significant cost reductions and new sales potentials to its users and providers. The advantages of using cloud computing include: i) reduced hardware and maintenance cost, ii) accessibility around the globe, and iii) flexibility and the highly automated process wherein the customer need not worry about software up-gradation which tends to be a daily matter.

As promising as it is, cloud computing is also facing many challenges like data security and data integrity that, if not well resolved, may impede its fast growth. Cloud computing is used by many software industries nowadays. Since the security is not provided in cloud, many companies adopt their unique security structure. The data placed in the cloud is accessible to everyone, security is not guaranteed. Cloud computing is used by many software industries nowadays. Since the security is not provided in cloud, many companies adopt their unique security structure. The data placed in the cloud is accessible to everyone, security is not guaranteed.

Now there was a problem that how to efficiently verify the correctness of the outsourced cloud data without the local copy of data files becomes a big challenge for data storage security in cloud computing. Downloading the data for verification is an expensive process. It makes the process much slower. To ensure security, cryptographic techniques cannot be directly adopted. Sometimes the cloud service provider may hide the data corruptions to maintain the reputation.

Later a critical exploration area for system protection, information access control has been advancing in the previous thirty years and conventional access control architectures more often than not expect the information proprietor and the servers putting away the information are in the same trusted space, where the servers are completely endowed as an omniscient reference screen in charge of characterizing and authorizing access control arrangements. This suspicion however didn't really holds in distributed computing subsequent to the information proprietor and cloud servers are liable to be in two distinct spaces. Encrypting data through certain cryptographic primitive(s), and disclosing decryption keys just to approved clients. This system really has been broadly embraced by existing works which go for securing data storage on untrusted servers.

In this paper, open testing issue like data security and data integrity are tended to and determined by permitting the data owner to appoint the vast majority of the computation tasks included in information access control to untrusted cloud servers without revealing the hidden information substance. We accomplish the objective by misusing and remarkably joining strategies encryption. Our proposed plan additionally has remarkable properties of client access benefit classifiedness and client mystery key responsibility. Data files are encrypted using public key components corresponding to their attributes.

II RELATED WORK

Sravan Kumar R and Ashutosh Saxena [1] have have worked to facilitate the client in getting a "proof of integrity of the data " which he wishes to store in the cloud storage servers with bare minimum costs and efforts.

Meiko Jensen et al. [2] have shown that to improve cloud computing security, the security capabilities of both web browsers and web service frame works, should be strengthened. This can best be done by integrating the latter into the former.

Armbust M Fox et al. [3] discuss that resources should be virtualized to hide the implementation of how they are multiplexed and shared. Kaufman, L. M. [4] have shown that to ensure CIA (Confidentiality, Integrity and Availability) of the information, the service provider should offer tested encryption schema, stringent access controls and scheduled data backups.

II BASIC MODELS AND GOALS

(i) System Models

We accept that the system is made out of the accompanying gatherings: the Data Owner, numerous Data Consumers, numerous Cloud Servers, and a

Third Party Auditor if important. To get to information records shared by the information proprietor, Data Consumers, or clients for quickness, download information documents of their enthusiasm from Cloud Servers and afterward unscramble. Neither the information proprietor nor clients will be constantly on the web. They come online just on the need premise. Starting now and into the foreseeable future, we will likewise call information documents by records for curtness. Cloud Servers are constantly online and worked by the Cloud Service Provider (CSP). The Third Party Auditor is additionally an online gathering which is utilized for evaluating each record access occasion.

(ii) Design Goals

Our primary design objective is to help the information proprietor to accomplish information security and information respectability of documents put away by Cloud Servers. In particular, we need to empower the information proprietor to implement an one of a kind access structure on every client, which correctly assigns the arrangement of documents that the client is permitted to get to. We additionally need to keep Cloud Servers from having the capacity to learn both the information record substance and client access benefit data. Also, the proposed plan ought to have the capacity to accomplish security objectives like client responsibility and bolster essential operations, for example, client stipend/renouncement as a general one-to-numerous correspondence framework would require.

IV OUR PROPOSED SCHEME

In order to achieve secure, versatile and fine-grained access control on outsourced information in the cloud, we use and exceptionally consolidate the accompanying progressed cryptograhphic systems. For every information record the owner assigns out an arrangement of important characteristics which are vital for access control. Different data files can have a subset of traits in like manner. Every quality is connected with a form number with the end goal of characteristic upgrade as we will talk about later. Cloud Servers keep a trait history list AHL which records the rendition advancement history of every quality and PRE keys utilized. Notwithstanding these important properties, we additionally characterize one fake quality, signified by image AttD with the end goal of key administration. AttD is obliged to be incorporated in each information document's property set and will never be overhauled. The entrance structure of every client is executed by an entrance tree. Inside hubs of the entrance tree are edge doors. Leaf hubs of the entrance tree are connected with information document characteristics. The fake trait won't be appended to whatever other hub in the entrance tree. Fig.1 delineates our definitions by a sample. Moreover, Cloud Servers additionally keep a client list UL which records IDs of all the legitimate clients in the framework. Fig.1 gives the depiction of documentation to be utilized as a part of our plan.

Notation	Description
PK, MK	system public key and master key
T_i	public key component for attribute i
t_i	master key component for attribute i
SK	user secret key
sk_i	user secret key component for attribute i
E_i	ciphertext component for attribute i
I	attribute set assigned to a data file
DEK	symmetric data encryption key of a data file
P	user access structure
L_P	set of attributes attached to leaf nodes of P
Att_D	the dummy attribute
UL	the system user list
AHL_i	attribute history list for attribute i
$rk_{i\to i'}$	proxy re-encryption key for attribute i from its current version to the updated version i'
$\delta_{O,X}$	the data owner's signature on message X

Fig. 1: Notation used in our scheme description

Presently we will explain the usage of abnormal state operations, i.e., System Setup, New File Creation, New User Grant, and User Revocation, File Access, File Deletion, and the communication between included gatherings. At calculation level, we concentrate on the execution of low level calculations that are summoned by framework level operations.

System Setup In this operation, the information proprietor picks a security parameter κ and calls the calculation level interface.

ASetup(κ), which yields the framework open parameter PK and the framework expert key MK. The information proprietor then signs every part of PK and sends PK alongside these marks to Cloud Servers.

New File Creation Before transferring a document to Cloud Servers, the information proprietor forms the information record as follows.

- select a unique *ID* for this data file;
- randomly select a symmetric data encryption key *DEK* $R\leftarrow K$, where *K* is the key space, and encrypt the data file using *DEK*;
- define a set of attribute *I* for the data file and encrypt *DEK* with *I* using KP-ABE, i.e., $(\tilde{E}, \{Ei\}i\in I) \leftarrow AEncrypt(I,DEK,PK)$.

New User Grant When a new user wants to join the system, the data owner assigns an access structure and the corresponding secret key to this user as follows.

- assign the new user a unique identity *w* and an access structure *P*;
- Generate a secret key *SK* for *w*, i.e., $SK \leftarrow AKeyGen(P,MK)$;
- encrypt the tuple $(P, SK,PK, \delta O,(P,SK,PK))$ with user *w*'s public key, denoting the ciphertext by *C*;
- Send the tuple $(T,C, \delta O,(T,C))$ to Cloud Servers, where *T* denotes the tuple $(w, \{j, skj\}\, j\in LP \setminus AttD)$.

On receiving the tuple $(T,C, \delta O,(T,C))$, Cloud Servers processes as follows.

- verify $\delta O,(T,C)$ and proceed if correct;
- store *T* in the system user list *UL*;
- forward *C* to the user.

On accepting C, the client first decodes it with his private key. At that point he checks the mark O,(P,SK,PK). In the event that right, he acknowledges (P, SK,PK) as his entrance structure, secret key, and the system public key. Cloud Servers store all the secret key parts of SK with the exception of the one comparing to the fake quality AttD.

Client Revocation we begin with the instinct of the client renouncement operation as takes after. At whatever point there is a client to be denied, the data owner first decides a negligible arrangement of traits without which the leaving client's entrance structure will never be fulfilled. Next, he redesigns these traits by rethinking their comparing system master key parts in MK. Public key segments of all these

upgraded qualities in PK are re-imagined in like manner. At that point, he redesigns client mystery keys as needs be for every one of the clients aside from the one to be repudiated. At last, DEKs of influenced information records are re-scrambled with the most recent rendition of PK. The primary issue with this instinctive plan is that it would present a substantial processing overhead for the information proprietor to re-scramble information documents and may require the data owner to be constantly online to give secret key upgrade administration to clients. All the more particularly, we isolate the client repudiation plan into two stages.

In the first stage, the data owner decides the insignificant arrangement of properties, reclassifies MK and PK for included characteristics, and creates the relating PRE keys. He then sends the client's ID, the insignificant property set, the PRE keys, the upgraded open key segments, alongside his marks on these segments to Cloud Servers, and can go logged off once more. Cloud Servers, on getting this message from the information proprietor, expel the denied client from the framework client list UL, store the upgraded open key parts and in addition the proprietor's marks on them, and record the PRE key of the most recent variant in the quality history list AHL for each overhauled trait. AHL of every quality is a rundown used to record the variant development history of this characteristic and in addition the PRE keys utilized. Each characteristic has its own AHL. With AHL, Cloud Servers have the capacity to figure a solitary PRE key that empowers them to redesign the trait from any chronicled form to the most recent rendition. When a client repudiation occasion happens, Cloud Servers simply record data put together by the information proprietor as is beforehand examined. If there is a record information access demand from a client, do Cloud Servers re-scramble the asked for documents and overhaul the asking for client's secret key.

Record Access: This is likewise the second phase of client disavowal. In this operation, Cloud Servers react client demand on information document get to, and upgrade client mystery keys and re-encode asked for information records if essential. As is portrayed in Fig. 3, Cloud Servers first confirm if the asking for client is a substantial framework client in UL. In the event that genuine, they overhaul this current client's mystery key segments to the most recent adaptation and re-scramble the DEKs of asked for information documents utilizing the most recent variant of PK. Eminently, Cloud Servers won't perform upgrade/re-encryption if mystery key parts/information documents are as of now of the most recent variant. At last, Cloud Servers send redesigned mystery key parts and in addition ciphertexts of the asked for information documents to the client. On accepting the reaction from Cloud Servers, the client first checks if the guaranteed form of every characteristic is truly more up to date than the present rendition he knows.

File Deletion This operation can only be performed at the request of the data owner. To delete a file, the data owner sends the file’s unique *ID* along with his signature on this *ID* to Cloud Servers. If verification of the owner’s signature returns true, Cloud Servers delete the data file.

V PERFORMANCE ISSUE

Security and Integrity: Proposed system provide data Integrity as follows: Before uploading a file , data owner select unique ID of the file then randomly select the encryption keys. Later a set of attributes are selected from file, those selected attributes are encrypted using previously selected keys. By following this steps both integrity and security is achieved.

VI CONCLUSION

In this paper, open challenging issue like data security and data integrity are addressed and resolved by allowing the data owner to delegate most of the computation tasks involved in data access control to untrusted cloud servers without disclosing the underlying data contents. We achieve the goal by exploiting and uniquely combining techniques encryption.

VII REFERENCES

[1] Sravan Kumar R and Ashutosh Saxena "Data Integrity Proofs in Cloud Storage" 978-1-4244-8953-4/11/$26.00 © 2011 IEEE.

[2] Meiko Jensen ,Jorg Sehwenk et al., "On Technical Security,Issues in cloud Computing "IEEE International conference on cloud Computing,2009.

[3] M.Jensen ,N.Gruschka et al., "The impact of flooding Attacks on network based services"Proceedings of the IEEE International conference on Availiabilty,Reliability and Security (ARES) 2008.

[4] Armbrust ,M. ,Fox, A., Griffth, R., et al "Above the clouds: A Berkeley View of Cloud Computing" , UCB/EECS-2009-28,EECS Department University of California Berkeley, 2009.

[5] Kaufman, L. M. (2009)."Data Security in the World of Cloud Computing." IEEE Security and Privacy 7(4): 61-64.

[6] Sloan, K. (2009)."Security in a virtualised world." Network Security 2009(8): 15-18.

[7] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Proc. of ESORICS '09, 2009

[8] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proc. of SecureComm'08. New York, NY, USA: ACM, 2008, pp. 1–10.

[9] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in Proc. Of Eurocrypt'03. Warsaw, Poland: Springer-Verlag, 2003, pp. 416– 432.

About Authors:

I am Venkata Subba Reddy.I have completed Btech from BVSR Engineering College, Chimakurthy, AP. I am purshuing my Mtech from Nova College of Engineering & Technology, Vijayawada. My research interest in data mining.

Mr. T.P V V Srinivasa Rao is a qualified person Holding M.Tech Degree in CSE from JNTU Kakinada, He is working as an Assistant Professor in NOVA College of Engineering Technology .He guided students in doing IBM

projects at NOVA ENGINEERING College. Who has Published 6 research Papers in various international Journals and workshops.

Dr. Srinivas Rao J received Ph D from CMJ University Meghalaya, M.Tech in Computer Science & Engineering from KL University in 2008. INDIA .He is an Outstanding Administrator & Coordinator. He is having 16 years of experience and handled both UG and PG classes. Currently he is working as a Director & Professor in NOVA College of Engineering Technology, Vijayawada, A.P, INDIA . He has Published 42 research Papers in various international Journals and workshops with his incredible work to gain the knowledge for feature errands.