ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 403- 407 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

An Empirical Model of Data Monitoring Service over cloud

[1]N.Sowjanya, [2]Ch. Swapna Priya
[1]Final M.Tech Student, [2]Assistant Professor
[1,2]Dept of Computer Science and Engineering
[1,2]Pydah College of Engineering, Visakhapatnam, AP, India

Abstract: Auditing over cloud data is an interesting research topic in cloud computing. Data owner uploads data fragments after the segmentation of data component to cloud server and auditor monitors data component which is uploaded by the data owner, traditional approach completely depends on third party auditor. In our work we propose an efficient auditing protocol with Meta data transfer to third party auditor instead of complete data component and dynamic updating when a block of data component corrupted.

INTRODUCTION

Cloud computing has been visualize the next generation architecture of IT endeavor due to its large list of advantages in the IT history: on demand service, location independent, resource pooling and rapid resource elasticity. From users side in clouding both individuals storing data distant into the cloud in easier on demand manner brings requesting benefits: relief of the burden of storage management global data access with dependent geo-graphical locations and reducing of large disbursement on hardware / software and personnel maintenances etc.

Present days cloud service is a frequently increasing technology due to its efficient features as a resource area and data storage area. It can be used as an application, an operating system or virtual machine and many advantages with cloud service technology. Cloud service provider follows pay and use relationship with clients and the data owner. They do not know where the real data is stored but he/she can surf the cloud when it required by verifying themselves with their authentication credentials.

Data Owner: Data Owner or User is a person stores more amount of data on server which is maintained by the service provider or the individual who is storing data or data component to the service provider. User has a privilege to upload their data on cloud without bothering about storage and maintenance. A service provider will provide services and privileges to the user. The major goal of cloud data storage is to achieve the exactness and probity of data stored in cloud.

Third Party Auditor: Third party auditors acts as verifier, verifies on users request for storage exactness and probity of data. This Auditor Communicates with Cloud Service Provider and monitors data components which are uploaded by the data owner.

The proposed work describe that user can browse the data stored in cloud as if the local one without bothering about the probity of the data. Therefore TPA is used to verify the probity of data. It maintains the privacy protecting public auditing. It verifies the probity of the data and the storage exactness. It also maintains data dynamics & batch auditing. The main benefits of storing data on a cloud is ease of burden in storage maintenance, global data access with location independent, reducing of large expenditure on hardware / software and self-maintenance.

Batch Auditing: It also maintains batch verifying through which efficiency is increasing. It allows TPA to process different thread parallel which reduces communication and calculation cost. Using this method we can find the invalid response. It uses Bi-Linear Signature BLS which is proposed by Boneh, Lynn and Shacham to achieve batch verifying.

Data Dynamics: It maintains data dynamics where user can rapidly update the data stored on a cloud service. It supports block level manipulations such as insertion, deletion and modification. Author of [2] proposed method which maintains parallel public verification capability and data dynamics. It uses Merkle Hash Tree works only on encoded data. It uses MHT for Cloud Service Provider who provide some sort of methods through which user will get the acknowledgement that cloud data is secure or is stored as the same. Through this alterations can be done and there will be no data loss. Organization provides different services to cloud users. Acquaintance and probity of cloud service data should be supported by cloud service provider. The service provider should protect user's data and applications are secured in cloud. Cloud service provider may not alter or access user's data. The Cloud Service Provider allows the Data Owner to upload the data items and allows Third Party Auditor to verify whether he/she is authenticated.

Cloud storage is an important service of cloud computing which allows data owners to change data location from their local computing systems to the cloud. More owners initiate to store the data in the cloud. This novel prototype of data deploying service also introduces novel security issues. Owners would distress that the data would be lost in the cloud. This is because of data loss could happen in any infra-structure, no matter what high degree of dependable parameters cloud service providers would take. The cloud service providers might be corrupt. They could throw away the data that have not been browsed or very few times accessed, to save the storage

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 403- 407 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

space and claim that the data still stored in the cloud. Therefore the owners required to be satisfied that the data are systematically stored in the cloud.

Deploying data into the cloud reduces the cost and complexity of big scale data storage and it does not action any agreement on data probity and availability. This complication is if not properly inscribed may block the successful hosting of the cloud architecture. As users no long time physically possess the storage of their data and the traditional cryptographic parameters for the ambition of data security protection process cannot be directly included. Therefore to maintain the exactness of deployed data without the regional copy of data files becomes a main task for data storage security in Cloud Computing. It is very simple for downloading the data for its probity verification is not a practical result due to the expensiveness in input or output cost and translating the file over the network. It is generally insufficient to find the data corruption when browsing the data might be very late for recovering the data loss or damage or corruption. Let us consider large size of the deployed data and the user's awkward resource capability and the ability to verify the exactness of the data in a cloud habitat can be dangerous and cost for the cloud users.

RELATED WORK

In the previous architectures data components can be uploaded by the data owners and the same data component can be forwarded to auditor to monitoring the data which is uploaded to the server. But it leads to privacy issue when data owner transfer entire data component to the auditor. So in this protocol we are proposing an efficient auditing protocol by forwarding entire data component to the third party auditor.

In cloud service data storage, users store their data in the cloud database and no longer possess the data locally. Therefore the exactness and availability of the data files stored on the cloud servers must be approved. One of the main issues is to detect any unauthorized data alteration and corruption and possibly due to server compromise and random Byzantine failures. In the decentralized case when such deviations are successfully recognize to find which server data error lies in and also of great implication and since it can always be the initial step to fast reveal the storage errors and finding possible threats of external attacks.

The simple Proof of retrievability method can be made using a keyed hash function as $h_k(F)$. In this method the verifier before achieving the data file F in the cloud storage and pre-computes the hash of F using $h_k(F)$ and stores hash result as well as the secret key as K. To verify the probity of the file F is lost the verifier publish the secret key K to the cloud achieve and query it to calculate and return value of $h_k(F)$. By storing various hash values for various keys the verifier verify for the probity of the file F for several times, each one being an independent proof.

A public auditing method consists of four algorithms such as Key_Generation, Sig_Generation, Gen_Proof, Verify_Proof.

Key_Generatiom is a key create method that is process by the user to setup the method.

Sig_Generation is processed by the user to create verification meta-data which consists of MAC, signatures and related information that will be used for verifying.

Gen_Proof is process by the cloud server to generate a proof of data storage.

Verify_Proof is process by the trust auditor to verify the proof from the cloud server.

Running a public verifying system consists of two steps such as Setup and Audit.

Setup: The user selects the public and secret tokens of the system by processing Key_Generation and pre-processes of the data file F by using Sig_Generation to trigger the verification Meta data at the cloud server and removes its regional copy. As part of pre-processing the user may modify the data file F by enlarging it or consisting additional metadata to be stored at server.

Audit: The TPA sends a verification message to the cloud service provider to make sure that the server has maintained the data file F correctly at the time of the verification. The cloud server will derivate a result message by processing Gen_Proof using F and its verification process of metadata as inputs. The TPA then verifies the results through Verify_Proof[14].

PROPOSED WORK

In this paper we propose an efficient auditing service with authentication, probity of data and security as primary factors in the architecture. The proposed work determines that user can access the data in cloud as without bothering about the exactness of the data. We improved the previous approach with efficient cryptographic method and secure authentication method. We also proposed dynamic block updating of corrupted block while intimated by the third party auditor.

Overviews of three roles are as follows

Data Owner (DO): He or she has the data files to be stored in the cloud database and depend on the cloud for data maintenance and it can be a customer or an organization. Data owner uploads the data components in the cloud.

Cloud Storage Service Provider (CSP): It provides data storage service and has enough storage space to maintain clients data and updates blocks if any corrupted over database. Cloud service provider allows an authorized auditor to monitor the data components and instant mails can be forwarded to Data owner.

Third Party Auditor (TPA): A trusted person who control or monitor data deployed by the data owner. Auditor

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 403- 407 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

receives initiation and authentication parameters and then monitors data components.

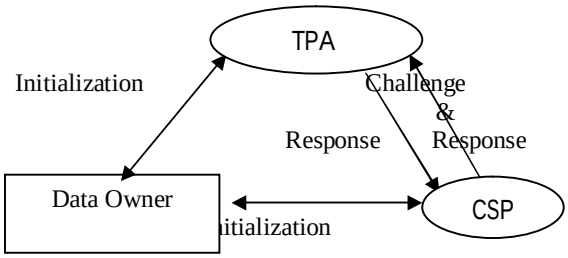

Fig 1: Auditing Architecture

In our method data owner apply signature generation method on each block of the data and creates the hash code and encrypts the content with Triple DES algorithm and uploads in to the server. Data Components sre divided into $m_1,m_2….m_n$ & generates random tag key set$(t_1,t_2…..t_n)$.

Every individual block can be encrypted with tag keys and then it forward the file meta data details and key to the third party auditor (verifier). There the auditor process same signature generation method and generates signature on the blocks and then verifies the both signatures if any block code is not matched that sends alert message to the data owner, then the administrator can forward only the revised information instead of total content then the user can browse the information which is given by the cloud service provider.

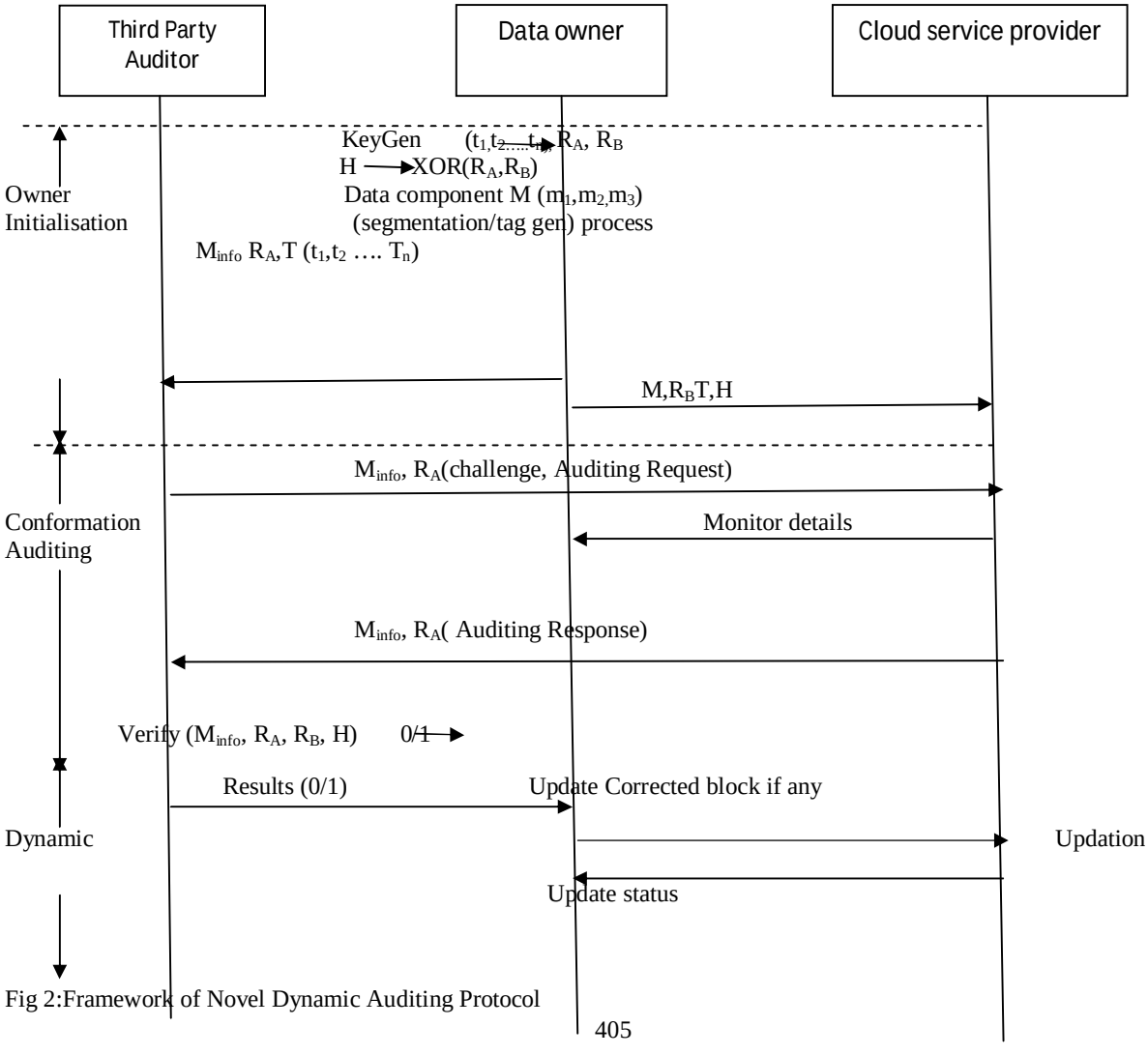

Fig 2:Framework of Novel Dynamic Auditing Protocol

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 403- 407 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

Table 1:Notations

Symbol	Meaning
M	Data component
T	Set of tag generation keys
R_A	Random challenge to Auditor (Large Prime Number)
R_B	Random Challenge to Cloud server (Large Prime Number)
$H(R_A,XOR, R_a)$	Hash code after XOR Over R_A and R_B
M_{info}	Meta or abstract informaton of M
n	Number of blocks in the each component

The above Figure shows entire architecture of the protocol, initially data owner segments the data component or file into number of blocks separated by a delimiter as space and generates a random tag key set which is required for encryption of individual blocks respectively. Data owner generates two random challenges for authentication of third party auditor at cloud service provider (CSP) while monitoring the data components of particular data owner. Data owner after encryption of data component uploads to the cloud storage area along with Tag key set and verification parameters and forwards initiation parameters to the auditor for monitoring of data component.

Step by Step Process for protocol Implementation:

Step1: Data owner fragments Data component D into n blocks (m_1,m_2….m_n).

Step2: Generates a random tag key set T (t1,t2…..tn) to encrypt the block with triple DES algorithm and finds signatures on encrypted blocks for authentication

Step3 : Generates random challenges R_A,R_B and computes hash value of xor between R_A and R_B.

x := hash (R_A XOR R_B)

Step4 : Forward Data component, Tag key set and RB to service provider and meta data and authentication parameters (M_{info} R_A,T (t_1,t_2 …. T_n)) to Auditor

Step5 : data owner Checks authentication by re-computing hash code with auditor RA.

Step6 : Auditor again divides D in ti number of blocks at server end, encrypts and applies same signature and compares signatures of corresponding blocks

Step7 : Monitoring Status can be forwarded t Data owner through smtp implementation

Step8: Auditor updates Data component status to the Data owner and updates the block if corrupted

Auditor receives the initiation parameters and meta data for monitoring of data component and authenticate himself at cloud service provider by forwarding the random challenge (R_A).Cloud service provider validates the auditor by generating the hash code of XOR (R_A,R_B),if authentication is success, cloud service provider allows the auditor to monitor the data component and instantly forward a mail response to the data owner. Data owner receives monitoring status from auditor, if uploaded data is same as monitored data then no issue otherwise data owner updates corrupted block which is informed by the auditor report.

CONCLUSION

We conclude that our work with an efficient auditing protocol without losing its data probity, In this approach, the user need not to forward the data components to the auditor directly, but auditing can be done efficiently. We can enhance our approach by increasing the authentication approach rather than simple random challenges. From the traditional approaches we are not entirely depend on the third party verifiers, therefore the protocol authorize the auditor to monitors data items of meta information only that provides the extracted information of data component. Data owner can receive the general monitoring details.

REFERENCES

[1] S. Marium, Q. Nazir, A. Ahmed, S. Ahthasham and Aamir M. Mirza, "Implementation of EAP with RSA for Enhancing The Security of Cloud Computig", International Journal of Basic and Applied Science,vol 1, no. 3, pp. 177-183,2012.

[2] Q. Wang, C. Wang,K.Ren, W. Lou and Jin Li "Enabling Public Audatability and Data Dynamics for Storage Security in Cloud Computing", IEEE Transaction on
Parallel and Distributed System, vol. 22, no. 5, pp. 847 859,2011.

[3] B. Dhiyanesh "A Novel Third Party Auditability and Dynamic Based Security in Cloud Computing" , International Journal of Advanced Research in Technology, vol. 1,no. 1, pp. 29 -33, ISSN: 6602 3127, 2011

[4] P. Mell and T. Grance, "The NIST definition of cloud computing,"National Institute of Standards and Technology, Tech. Rep., 2009.

[6] T. Velte, A. Velte, and R. Elsenpeter, Cloud Computing: A Practical Approach, 1st ed. New York, NY, USA: McGraw-Hill, Inc., 2010, ch. 7. Stoica, and M. Zaharia, "A view of cloud computing," Commun. ACM,

[7] L. N. Bairavasundaram, G. R. Goodson, S. Pasupathy, and J. Schindler, "An analysis of latent sector errors in disk drives," in SIGMETRICS, L. Golubchik, M. H. Ammar, and M. Harchol-Balter, Eds. ACM, 2007, pp. 289–300.

[8] B. Schroeder and G. A. Gibson, "Disk failures in the real world: What does an mttf of 1, 000, 000 hours mean to you?" in FAST. USENIX, 2007, pp. 1–16.

[7] M. Lillibridge, S. Elnikety, A. Birrell, M. Burrows, and M. Isard, "A cooperative internet backup scheme," in USENIX Annual Technical Conference, General Track.USENIX, 2003, pp. 29–41.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 403- 407 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

[9] Y. Deswarte, J. Quisquater, and A. Saidane, "Remote probity checking," in The Sixth Working Conference on Probity and Internal Control in Information Systems(IICIS). Springer Netherlands, November 2004.

[10] M. Naor and G. N. Rothblum, "The complexity of online memory checking," J. ACM, vol. 56, no. 1, 2009.

[11] A. Juels and B. S. K. Jr., "Pors: proofs of retrievability for large files," in ACM Conference on Computer and Communications Security, P. Ning, S. D. C. di Vimercati, and P. F. Syverson, Eds. ACM, 2007, pp. 584–597.

[12] T. J. E. Schwarz and E. L. Miller, "Store, forget, and check: Using algebraic signatures to check remotely administered storage," in ICDCS. IEEE Computer Society, 2006, p. 12.

[13] D. L. G. Filho and P. S. L. M. Barreto, "Demonstrating data possession and uncheatable data transfer," IACR Cryptology ePrint Archive, vol. 2006, p.150, 2006.

[14] F. Seb´e, J. Domingo-Ferrer, A. Mart´ınez-Ballest´e, Y. Deswarte, and J.-J. Quisquater, "Efficient remote data possession checking incritical information .

[15] Cong Wang, Sherman S.M, Qian Wang, Kui Ren, Wenjing Lou "Privacy-Preserving Public Auditing for Secure Cloud Storage".