Enhancement in Security of Data Retrieval in Cloud Storage Using Cloud Auditing and CKS Cryptographic Technique

Madhuri Mishra[1], C. P. Singh [2]

[1]M.Tech Computer Science, College of Science & Engineering, Jhansi, U.P. India
[2]Asst. Professor (CSE Dept.), College of Science & Engineering, Jhansi, U.P. India
[1]madhurimishra.16@gmail.com[2]cp.singh1984@gmail.com

ABSTRACT

The security and authentication of data storage in cloud network are major issues. For the authentication and security of data used third party auditor. The third-party auditor authenticates the user and cloud service provider. The third-party auditor provides the authentication process of cloud data center to the user level. The authentication of user and data center provide the facility of cloud data auditing. The cloud auditing of data used the process of proof of data retrieval, for the data retrieval over the cloud network. In the process of data auditing required security constraints for the integrity of data. For the integrity of data, various authors used various cryptography techniques symmetric and asymmetric. In this paper proposed cyclic key based data security technique for the integration of cloud data auditing. The proposed method is implemented in ASP.NET with SQL server.

Keyword: - Cloud Computing, security, integrity, TPA, CKS

1. INTRODUCTION

The cloud security issue damages the data and faced a problem of authorization and authentication. For the improvement of cloud data security, various cryptography techniques are used. All the cryptography technique suffered a problem of encryption and decryption time. How to reduce the encryption and decryption time it is big question for the reduction of computational time and accelerator the process of key distribution and key allocation is the major issue in cloud data storage. It investigates the problem of ensuring the security and dependability for cloud data storage under the aforementioned adversary model [1, 2]. In particular, we aim to design efficient mechanisms for dynamic data verification and operation. The personal computer shifted processing closer to the user but as communication bandwidth increased the advantages of remote server provision re-emerged. The Internet had always provided some remote access but increasing bandwidth made it necessary to consider computing beyond firewall protected local administrative domains, giving rise to new security concerns. Web-based, Service-Oriented Architectures took the provision of computing to a global scale. Users should be able to plug in and do their computing work with little or no attention to how the distributed computing is actually orchestrated. While grid technologies were popular for scientific infrastructure, they did not have the great commercial impact. Cloud computing gained significant momentum with widespread user adoption of dynamic websites (e.g. for e-commerce)[3,4,5]. These were typically hosted on servers with PC-compatible architectures and were decoupled from infrastructure, as the development and deployment of high-efficiency PC hardware virtualization surged. Cloud service offerings are typically divided into three broad categories: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). In terms of security, the operating systems and software running on the VMs generally need to be managed no differently than on physical, dedicated servers. The exception to this is "para-virtualized" device drivers that are installed into the VMs to increase efficiency. These drivers are necessarily aware of running in a VM. Instead of using expensive emulated device access they typically interact directly with

the VM host via some agreed channel. However, there is little that an IaaS user can do other than trusting the para-virtualized device driver authors, or choosing to use much slower virtual hardware via native device drivers. The rest of paper is as in section 2 discuss the Public Auditability. In section 3 discuss proposed Work. In section 4 discuss the experimental result and analysis. Finally, discuss conclusion & future work in section 5.

2. PUBLIC AUDITABILITY

This kind of auditability allows anyone, not just the client, to challenge the cloud server and perform data verification check. This is where a Third Party Auditor (TPA) comes into play. The Public audit allows Third Party Auditor along with customer to examine the integrity of the contracted details saved on reasoning & Privacy Preserving allows Third Party Auditor to do the audit without inquiring for the local duplicate of the details [7, 9]. Through this plan, Public auditability also allows clients to delegate the integrity verification tasks to Third Party Auditor while they themselves can be unreliable or not be able to commit necessary computation resources performing continuous verifications [1]. Public auditability allows anyone, not just the client (data owner), to challenge the cloud server for the correctness of data storage while keeping no private information. Then, clients are able to delegate the evaluation of the service performance to an independent third party auditor Third Party Auditor, without the devotion of their computation resources [5]. In the cloud, the clients themselves are unreliable or may not be able to afford the overhead of performing frequent integrity checks. Thus, for practical use, it seems more rational to equip the verification protocol with public auditability, which is expected to play a more important role in achieving economies of scale for Cloud Computing. Homomorphism authenticators are unforgettable verification metadata generated from individual data blocks, which can be securely aggregated in such a way to assure an auditor that a linear combination of data blocks is correctly computed by verifying only the aggregated authenticator.

3 PROPOSED SYSTEM

A. Participants

In the proposed scheme to provide secure data storage, three participants are involved. These participants are Data Owner (DO), Third Party Auditor (TPA) and Cloud Service Provider (CSP). The role of these participants is described in

Table I ROLE OF PARTICIPANTS

Participant	Role
Data Owner (DO)	Data owner is a person who utilizes the Data Owner (DO) storage services provided by the cloud service provider.
Third Party Auditor (TPA)	TPA checks the integrity of the data stored on cloud.
Cloud Service Provider (CSP)	CSP provides the storage services to the users

1. Admin section; - admin section deals with the user and TPA data authentication mechanism. The admin gives the whole ownership for the management of data security of user data. Here admin plays a role of CSP.
2. TPA: - Third-party auditor audits the user data with security constraints for the processing of submission of data and maintains the integrity of data. TPA also never changes and edits the content of data over the cloud server.
3. Data Owner: - The Owner submits the data to the cloud server with authentication of the key.

B. Cloud security

A cloud user stores their sensitive data to the cloud. Thus it becomes cloud service provider's responsibility to establish the secure mechanism. It should provide secure channels for sending and receiving data and also for storing data. Thus security is a major concern to cloud service providers. Service providers use different encryption-decryption algorithms for secure communication and storage of data. Algorithms like AES, RSA, DES, and SHA are most popular and secure among them. All the cryptography technique suffered a problem of encryption and decryption time. How to reduce the encryption and decryption time it is big question? All the asymmetric encryption algorithms are too slow and all symmetric encryption algorithms are too fast but not all are secure. "A Study of Encryption Algorithms AES, DES and RSA [9]

Table 2: Comparisons of DES, AES and RSA of Encryption and Decryption Time

S.NO	*Algor*	*Pack Size (KB)*	*Encrypt Time (Sec)*	*Decrypt Time (Sec)*
1	DES	153	3.0	1
	AES		1.6	1.1
	RSA		7.3	4.9
2	DES	118	3.2	1.2
	AES		1.7	1.2
	RSA		10.0	5.0
3	DES	196	2.0	1.4
	AES		1.7	1.24
	RSA		8.5	5.9
4	DES	868	4.0	1.8
	AES		2.0	1.2
	RSA		8.2	5.1
5	**DES**	312	3.0	1.6
	AES		1.8	1.3
	RSA		7.8	5.1

So In proposed research work, we have used AES Rijndael block cipher and SHA1 algorithms and a new methodology is used for generating CKS for checking integrity. Any cloud service provider should provide authenticity, integrity and availability to their cloud users.

Enciphering with AES Rijndael The Rijndael cipher is an iterative block cipher. It, therefore, consists of a sequence of transformations to encipher or decipher the data. Rijndael encryption and decryption begin and end with a step to mix subkeys with the data block. This extra step is done as a protection against cryptanalysis. To encipher a block of data in Rijndael, you must first perform an Add Round Key step (XORing a subkey with the block) by itself, then the regular transformation rounds, and then a final round with the Mix Column step omitted. Here encryptor is used of block size of 128 bits, key size of 16 bytes, cipher mode-CBC (cipher block chaining), input block size of 16 bytes and output block size of 16 bytes

SHA-1 (short for *Secure Hash Algorithm*) is one of several Cryptography hash function SHA-1 is most often used to verify that a file has been unaltered. This is done by producing a checksum before the file has been transmitted, and

then again once it reaches its destination. The transmitted file can be considered genuine only if **both checksums are identical**. Here hash size 160bits and the iteration count of 100.

C. Proposed Mechanism

1) *Key Generation*:

Data Owner uses AES Rijndael block cipher algorithm for generating the secret key, Rankey using a random number which is occurred with the uploaded file and CKS.

RandomNo=CalculateRandomNo()

RanKey=*Encryption[RandomNo]

Cyclic [File_Size_Char]=SplitFileSize[File_Size]

CyclicString=CyclicShiftBitWiseXOR(Cyclic[File_Size_Char])

CyclicKeyShift(CKS)=*Encryption [CyclicString]

StaticKey=GetStaticKey()

*Encryption:

PlainText=GetPlaneText()

Salt=CalculateSalt(Length(StaticKey))

secreteKey=GetKey(StaticKey,Salt)

Encryptor=RijndaelCipherEncryptor(Key[16], IV[16])
This generates a new key and initialization vector (IV) of 16 bytes
CryptoStream=GetCryptoStream(plaintext,Encryptor)

EncrytpedData=GetCipherString(CipherBytes(Crypto-Stream))

*Decryption:

EncrytpedData=GetEncryptedText()

Salt=CalculateSalt(Length(StaticKey))

secreteKey=GetKey(StaticKey,Salt)

Dencryptor=RijndaelCipherEncryptor(Key[16], IV[16])
CryptoStream=GetCryptoStream(EncryptedData,Decryptor)
DecryptedCount=GetCryptoStream(EncryptedData,Decryptor,Encrypteddata_length)
DecrypteData=GetString(DecryptedCount)

2) *Encryption:*

Firstly, data owner encrypt the file (F) using the secret key and then generate the hash of an encrypted file. A random number is generating with the selected file and then encryption of this creates Rankey.

CKS is encryption of cyclic string which is generated using file_size_length.

Here the algorithms involved in proposed scheme are shown.

Algorithm1:
DataOwner(DO):
1) Select file and generate associated random number (F, rno)

2)Encrypt file (F'),

3)generate Rankey, CSK

4)DO ⟶ CSP: (F')

5)CSP: store(F')

Algorithm 2

1) DO request for the file

2) CSP send the file(F') to TPA requested by DO

3) TPA send request for key to DO

4) DO send static key, CSK and Rankey of (F)

5) TPA get random number after decrypt Rankey

6)check random number is associated with the

requested file(F')

7) check the modification of the file(F') using CSK

8) TPA calculate cyclicString of cloud-stored file (F')

and compared to decrypted CKS(F)

9)if both are equal then

10)TPA send a report of verify=yes, modification=no,

data authentication, and Integrity is maintained

4 RESULTS AND ANALYSIS

An application has been designed and implemented in asp.net, vs 2015 framework and c# language on the network (LAN), to achieve the functionalities of the data Owner, TPA and cloud server. We have assumed that the cloud server, TPA and the owner are in the same system domain and sharing the uniform system parameters. Through this application the file can be transferred between these entities and the required result has been achieved.

5. CONCLUSION AND FUTURE WORK

Data integrity and isolation protections are put in place to mitigate the risks, users pose to one another in terms of data loss, for ensuring the data dynamic in this model design new protocol of key generation based on cyclic shift key generation technique. The cyclic key generation technique based on XOR operation of the binary key and provide secure session key. The analysis and evaluation have enabled us to draw some conclusions. Our proposed key generation demonstrates how integrity verification can be done with the just transfer of few bytes and offline execution of necessary algorithms. It also offers secure access control, managing access rights mechanism, audit trail, better performance and reduced overhead.

REFERENCES

[1] Jin Li, Xiao Tan, Xiaofeng Chen, Duncan S. Wong, FatosXhafa, "OPOR: Enabling Proof of Retrievability in Cloud Computing with Resource-Constrained Devices" IEEE 2015, Pp 195 205

[2] Qian Wang, KuiRen, Member, Wenjing Lou, Jin "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing" IEEE 2011 847 -859

[3] MeeraChheda, AnmolAchhra, PriyankaVaswani, RajeshwariAgale, VidyaBhise. "Public Auditing For The Shared Data In The Cloud". International Journal of Advance Foundation and Research in Computer (IJAFRC) 2015 Pp 724-728.

[4] Prof. N.L. Chourasiya, DayanandLature, ArunKumavat, VipulKalaskar, SanketThaware."Privacy-Preserving Public Auditing for Secure Cloud Storage" International Journal of Engineering Research and General Science, 2015Pp 744 -748.

[5] R.Guruprasath, M.Arulprakash "Privacy Preserving Public AuditingFor Shared Data With Large Groups In The Cloud" Journal of Recent Research in Engineering and Technology 2015 Pp 40-46

[6] MrunaliPingale, Prof. JyotiPingalkar"Security Preserving Access Control Mechanism In Public Clouds Using PANDA Security Mechanism" iPGCON, 2015 Pp 1-5.

[7] PradnyaChikhale, NamrataDwivedi, ParnaDutta, AparajitaSain, VrundaBhusari"Enhancing Data Storage Security In Cloud Computing Using PDDS Technique" PISER 2014 Pp 53-59.

[8] J.Aparna, Mr.R.Sathiyaraj"Auditing Mechanisms for Outsourced Cloud Storage" International Journal of Computer Science and Mobile Computing, 2014, Pp 219-229

[9] Dr. PrernaMahajan&Abhishek "A Study of Encryption Algorithms AES, DES and RSA for Security"Sachdevahttps://globaljournals.org/GJCST_Volume13/4-A-Study-of-Encryption-Algorithms.pdf

[10] Betzy K. Thomas, M. NewlinRajkumar "A Dynamic Public Auditing SecurityScheme To Preserve Privacy In Cloud Storage" IJSHJE 2013 Pp 93-97.

[11] GuangyangYang, Hui Xia, WentingShen, XiuxiuJiang,Jia Yu "Public Data Auditing with Constrained Auditing Number for Cloud Storage" 2015 IJSIA Pp 21-32.

[12] JianYang,Haihang Wang, Jian Wang, Chengxiang Tan , Dingguo "Provable Data Possession of Resource-constrained Mobile Devices in Cloud Computing" Journal Of Networks, 2011 Pp1033-1040.

[13] JavedAkthar Khan, RitikaArora"A Review of Cloud Environment and Recognition of Highly SecurePublic Data Verification Architecture using Secure Public VerifierAuditor" International Journal of Electrical, Electronics and Computer Engineering 2014 Pp144-148.

[14] HarleenKaur, Er. VinayGautam "A Survey of Various Cloud Simulators" International Journal of Computer Sciences and Engineering 2014 Pp35- 38.

[15] Clementine Gritti, Willy Susilo, Thomas Plantard ,Rongmao Chen "Improvements on Efficient Dynamic Provable Data Possession scheme with Public Veri_ability and Data Privacy"Centre for Computer and Information Security Research 2014 Pp 1-19.

[16] ChunmingGao, Noriyuki Iwane "A Social Network Model With Privacy Preserving And Reliability Assurance And Its Applications In Health Care " International Journal Of Energy, Information And Communications 2015, Pp.45-58.

[17] Mohammad Iftekhar Husain Steve KoAtriRudraSteveUurtamo "Almost Universal Hash Families Are Also Storage Enforcing " Department Of Computer Science And Engineering,University At Buffalo 2012 Pp 1-18.

[18] ChintalMaisheri, Deepak Sharma" Enabling Indirect Mutual Trust For Cloud Storage Systems ". International Journal Of Computer Applications 2013 Pp 1-11

[19] Frank Hans-Ulrich Doelitzscher "Security Audit Compliance For Cloud Computing" Plymouth University ,Thesis 2014

[20] B. Krishna Kumari, S. Swapna" Stability Based Service For Secure Cloud Storage " IJITECH, 2015 Pp0399-0403.