INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN : 2454-9924

Dynamic & Flexible Group Key Generation For Shared Dynamic Data In Cloud With User Revocation

Arthi P#1, Jagan A*2
*#1 P.G. Scholar, *2Assistant Professor*
Department of CSE,
Surya Group of Institution,
Vikravandi, Villupuram.

ABSTRACT:

Cloud computing has the number of characteristic use; among those scalability is the one which adapt to the increase in data's. The Data's where separated into block where each block was signed with a specified signature. cloud computing makes storage outsourcing become a rising trend, which promotes the secure remote data auditing and efficient public data integrity auditing for shared dynamic data. It provides an efficient public integrity auditing scheme with generation of minimized key using Elliptic Curve Cryptographybased onsecure group user revocation. Our scheme supports the public checking and efficient user revocation and also some nice properties, such as confidently, efficiency, countable and traceability of secure group user revocation. Finally, the security and experimental analysis show that compared with its relevant schemes our scheme is also secure and efficient.

Keywords: user revocation, data auditing, cloud computing.

1 INTRODUCTION:

In today's Computing world Cloud computing is one of the biggest innovative techniques that uses advanced computational power and it maximizedata sharing and data storing capabilities[1][2]. It has maximum storing capability on comparing with others. The issues of data integrity, data privacy and data access by unauthorized users are the main problem in cloud computing. The TTA (Trusted Third Party) or anyone user from the group will do the verification process in cloud. The Trusted Third Party is the one who will provide the verification process and check the data's was correct or not[8][9][10]. This process was termed as the Auditing. The trusted third party will also share and store the data's in the cloud computing.

Updating of data, Modification and sharing of data is kind of straightforward as a group. To verify integrity of the shared data, members within the cluster must calculate signatures on all shared datablocks. Completely different blocks in shared dataarea unite usually signed by completely different users owing to data modifications performed by completely different users.

User revocation is one of the largest security threats in data sharing in teams. During user revocation shared data block signed by revoked user must transfer and re-sign by existing user. This task is very inefficacious owing to the big size of shared data blocks on cloud[12]. Panda and is that the new public auditing mechanism for the maintaining integrity of shared data with economical user revocation within the cloud.

This mechanism relies on proxy re-signatures concept that permits the cloud to re-sign blocks on behalf of existing users throughout user revocation, so that downloading of shared data blocks isn't needed. Panda Plus is that the public auditor that audits the integrity of shared data while not retrieving the complete data from the cloud. It also monitor batch to verify multiple auditing tasks simultaneously.

Batch auditing done as number of process per second. The main characteristic that was taken into account is Correctness of data, Efficient and Secure User Revocation, Public Auditing, and Scalability. This study work comprises the ECC Algorithm, proxy re-signature, Batch Auditing process, and Comparative study.

The development of cloud computing motivates enterprises and organizations to outsource their data to third-party cloud service providers (CSPs), which will improve the storage limitation of resource constrain local devices[14][15][16]. Recently, some commercial cloud storage services, such as the simple storage service (S3) on-line data backup services of Amazon and some practical cloud based software Google Drive, Dropbox, Mozy, Bitcasa, and Memopal, have been built for cloud application. Since the cloud servers may return an invalid result in some cases, such as server hardware/software failure, human maintenance and malicious attack, new forms of assurance of data integrity and accessibility are required to protect the security and privacy of cloud user's data[16][17]. To overcome the above critical security challenge of today's cloud storage services, simple replication and protocols like Rabin's data dispersion scheme are far from practical application

The later protocols ensure the availability of data when a quorum of repositories, such as k-out-of-n of shared data, is given[15]. However, they do not provide assurances about the availability of each repository, which will limit the assurance that the protocols can provide to relying parties. For providing the integrity and availability of remote cloud store, some solutions, and their variants have been proposed[11][12]. In these solutions, when a scheme supports data modification, we call it dynamic scheme, otherwise static one (or limited dynamic scheme, if a scheme could only efficiently support some specified operation, such as append). A scheme is publicly verifiablemeans that the data integrity check can be performed not only by data owners, but also by any third-party auditor[20][21].

2 PROBLEM STATEMENT:

The major challenge is that, Group users consist of a data owner and a number of users who are authorized to access and modify the data by the data owner. The cloud storage server is semi-trusted, who provides data storage services for the group users. TPA could be any entity in the cloud, which will be able to conduct the data integrity of the shared data stored in the cloud server. Also, he/she shares the privilege such as access and modify (compile and execute if necessary) to a number of group users. The TPA could efficiently verify the integrity of the data stored in the cloud storage server, even the data is frequently updated by the group users. The data owner is different from the other group users, he/she could securely revoke a group user when a group user is found malicious or the contract of the user is expired.

3 THE SYSTEM ARCHITECTURE:

The entire cloud servers will maintain varied data's and it also maintain the user access permission. The registered users will upload the files into the proxy. The data manager will maintain the user details along with it, where the file also been keep on tracking. Each and every file was uploaded with a public key.

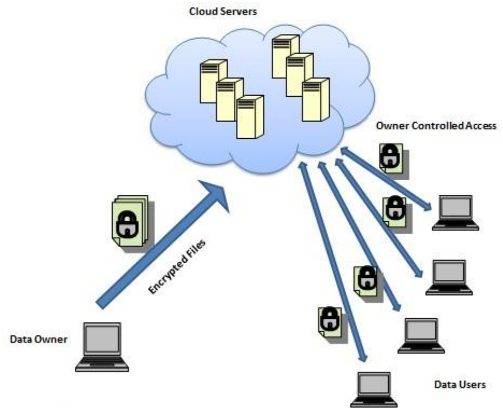

Figure 1. The System Model with Cloud storage, Data owner, and Data users.

4 TECHNIQUE:

These involve the following steps as the processing: Authentication, group manager, Store data into the cloud, Access control list, maintain revoke list, auditor verification.

Proxy re-signature: Proxy re-signature scheme is the process of signing the block by proxy which was already signed by the user. The proxy signing will be carried out based on the user behavior. If anyone from the team misbehaves or withdraw the proxy server will generate a signature for the entire team member. So, that security for the data has been provided by attaching new signature. These signatures may also subject to change when there is new upload of data to the proxy.

Authentication: In this module User want to register the personal details in the database and get the authentication processes to go forward. In this module User want to give the database to admin all the registration process is done by admin. After the registration process completed User can get the authentication permission, by using username and password login website. If the user enters a valid username/password combination they will be granted to access data. If the user enter invalid username and password that user will be considered as unauthorized user and denied access to that user.

Group manager: In this module the authorized user pay for the cloud server and get the allocation space. The Authorized person stores secure data to the cloud. Get the access control and then provide the group members.The group manager will act as a admin which has the full access permission on files that was uploaded into the cloud. It can modify the data in the cloud. The group manager provide the access control to the authorized user only.

Store data into the Cloud: Only the authorized user will store the data into the cloud. And the registered user will access the data. The cloud manager provides the permission for the specific user who needs to access the data. The user will access the data the file by using their signature and the private key provided by the group manager.

Maintain Revocation List: The cloud storage gets the resignation user details from the company and maintains the revocation list. Because identify the unauthorized users and they cannot access the cloud resource at any time, and revoked users will be incapable of using the cloud again once they are revoked.

5 CONSTRUCTION OF KEY GENERATION:

Scheme Details: LetG_a and G_b be two groups of order p, g be a generator of G_a, e: G_ax G_b⟶ G_b be a bilinear map, w be another generator of G_a. The global parameters are (y,p, G_a, G_b, g, w, H), where H is a hash function with H:{0,1}*⟶ G_a.

KeyGen. Given global parameters (y,p, G_a, G_b, g, w, H), a user v^A selects a random aϵZ_p, and outputs public key pk^A=ga and private key sk^A=a.

Rekey: The proxy generates a re-signing key rk^A→B as follows: (1) the proxy generates a random rϵZ_p and sends it to user v^A; (2) user v^Acomputes and sends r/a to user v^B, where sk^A=a; (3) user v^B calculates and sends rb/a to the proxy, where sk^B=b; (4) the proxy recovers rk^A→B= b/a ϵZ_p.

Sign. Given private key sk^A=a, block mϵZ_p and block identifier id, user v^A outputs the signature on block m as:

$$\sigma = (H(id)\omega^m) \text{ ϵ Ga.}$$

Resign. Given re-signing key rk^A→B, public key pk^A, signature σ, block m ϵ Z_p and block identifier id, the proxy checks that Verify (pk^A, m, id, σ) =1. If the verification result is 0, the proxy outputs perpendicular or else it will produce the output.

Verify. $pk^A$is the given public key, block m, block identifier id, and signature σ, a verifier outputs 1 if

$$\text{E}(\sigma, g) = e(H(id)w^M, pk_A),$$

and 0 otherwise.

6 COMPARATIVE STUDY:

On comparing the four key generating algorithm the Elliptic curve Cryptography with the Diffie-Hellman Algorithm, Diffie-Hellman algorithm have the process of key exchange and it can't be used for encryption, decryption, generation of digital signature. Even though RSA algorithm have the property of all the three: encryption, decryption, digital signature, key exchange it has large key size. The DSS has only Digital signature property whereas the Diffie-Hellman have a key exchange property but there is no encryption, decryption and digital signature property.

Algorithm	E/D*	DS*	KX*
RSA	Yes	Yes	Yes
ECC	Yes	Yes	Yes
DSS	No	Yes	No
Diffie-Hellman	No	No	Yes

E/D* - Encryption/Decryption,
DS* - Digital Signature,
KX* - Key Exchange.

7 RELATED WORK:

Starting solutions to the PDP problem were providedby M. Armbrust et al. [12] and G. Ateniese et al. [10]. Both useRSA-based hash functions to hash the entire file at everychallenge. This is clearly prohibitive for the serverwhenever the file is large.Similarly, Y. Zhu et al. [18] give a protocol for remotefile integrity checking, based on the Diffie-Hellmanproblem in composite-order groups. However, theserver must access the entire file and in addition theclient is forced to store several bits per file block, sostorage at the client is linear with respect to the numberof file blocks as opposed to constant.B. Wang, L. Baochun [23] propose a scheme that allowsa client to verify storage integrity of data acrossmultiple servers. However, even in this case, the servermust access a linear number of file blocks per challenge.D. Catalano et al. [22] first proposed the concept of enforcementof storage complexity and provided efficientschemes. Unfortunately the guarantee they provide isweaker than the one provided by PDP schemes since itonly ensures that the server is storing something at leastas large as the original file but not necessarily the fileitself.Provable data possession is a form of memory checking[25, 23] and in particular it is related to the conceptof sub-linear authentication introduced by B. Wang, L. Baochun [23]. However, schemes that provide sublinearauthentication are quite inefficient.Compared to the PDP scheme in [7], our scheme issignificantly more efficient in both setup and verificationphases since it relies only on symmetric-key cryptography.The scheme in [7] allows unlimited verificationsand offers public verifiability (which we do notachieve). However, we showed that limiting the numberof challenges is not a concern in practice. In addition,our scheme supports dynamic operations, whilethe PDP scheme in [7] works only for static databases.Compared to the POR scheme [8], our scheme providesbetter performance on the client side, requiresmuch less storage space, and uses less bandwidth (sizesof challenges and responses in our scheme is a smallconstant, less than the size of a single block). To appreciatethe difference in storage overhead, consider that,POR needs to store s sentinels per verification, whereeach sentinel is one data block in size (hence, a total ofs_t sentinels). In contrast, our scheme needs a singleencrypted value (256 bits) per verification. Note that, inPOR, for a detection probability of around 93%, whereat most 1% of the blocks have been corrupted, the suggestedvalue s is on the order of one thousand [8]. Like[7], POR is limited to static data (in fact, [8] considerssupporting dynamic data an open problem). In summary,we provide more features than POR by consumingconsiderably less resources 5.

To further compare the two schemes we assess theirrespective tamper detection probabilities. In [8], theoutsourced data is composed of d blocks and there ares sentinels. We compute the probability that, m beingthe number of corrupted blocks, the system consumesall sentinels and corruption is undetected. Similarly, We have recently learned that a MAC-based variant of our firstbasic scheme was mentioned by Ari Juels during a presentation atCCS '07 in November 2007 ([13]). This is an independent and concurrentdiscovery. Indeed, an early version of our paper was submittedto NDSS 2008 in September 2007.Evaluate our scheme, for the same amount of data, weassume t tokens, each based on r verifiers.For our scheme, the probability of no corruptedblock being detected after all t tokens are consumed is:

$$\frac{\frac{d-n}{m}}{(d,m)}$$

Note that h accounts for the number of different blocksused by at least one token. Indeed, to avoid detection,SRV cannot corrupt any block used to computeany token. It can be shown that, for a suitably larged, for every t, it holds that: n >minof(tr/2,d/2) withoverwhelming probability. Hence, the upper bound forEquation 2, when tr/2 < d/2, is given by:

$$\frac{\frac{d-tr}{m}}{(d,m)}$$

In [8], SRV can avoid detection if it does not corruptany sentinel block. Since a total of s*t sentinels areadded to the outsourced data, the probability of avoidingdetection is:

$$\frac{(d,m)}{\frac{d+(s*t)}{m}}$$

The above equations show that, in our scheme, for agiven t, increasing r also increases the tamper-detectionprobability. However, r has no effect on the total numberof tokens, i.e., does not influence storage overhead.As for POR, for a given t, increasing tamper detectionprobability requires increasing in s, which, in turn, increasesstorage overhead.

8 FUTURE ENHANCEMENT:

In Future the primitive of verifiable database with efficient updates is an important way to improve secure and verifiable outsourcing of cloud storage. A scheme efficient and secure data integrity auditing for share dynamic data with multi-user modification with user revocation are adopt to achieve the data integrity auditing of remote data and generate details and view about revoked user. This produce the efficient and dynamic key generation for the data's with the user revocation in the cloud.

9 CONCLUSION:

Thus, the ECC provide faster key generation which is easier to attach with the data parts. This will reduce the computational and communication cost and also the public Auditing is simpler when the key size is smaller.We obtain a signature of length 154 bits where breaking the scheme reduces to solving the Diffie-Hellman problem in a finite field of size approximately 2^{923}. An open problem that would enable us to get even better security while maintaining is the same length signatures. We hope future work on constructing elliptic curves or higher genus curves will help in solving this problem.

REFERENCES:

[1] Amazon. (2007) Amazon simple storage service (amazon s3).Amazon. [Online]. Available: http://aws.amazon.com/s3/.

[2] Google. (2005) Google drive. Google. [Online]. Available: http://drive.google.com/

[3] Dropbox. (2007) A file-storage and sharing service. Dropbox. [Online]. Available: http://www.dropbox.com/

[4] Mozy. (2007) An online, data, and computer backup software. EMC. [Online]. Available: http://www.dropbox.com/

[5] Bitcasa. (2011) Inifinite storage. Bitcasa. [Online]. Available: http://www.bitcasa.com/

[6] Memopal. (2007) Online backup. Memopal. [Online]. Available: http://www.memopal.com/

[7] M. A. et al., "Above the clouds: A berkeley view of cloud computing," Tech. Rep. UCBEECS, vol. 28, pp. 1–23, Feb. 2009.

[8] M. Rabin, "Efficient dispersal of data for security,"Journal of the ACM (JACM), vol. 36(2), pp. 335–348, Apr. 1989.

[9] J. G. et al. (2006) The expanding digital universe: A forecast of worldwide information growth through 2010. IDC. [Online]. Available: Whitepaper

[10] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at

untrustedstores," in Proc. of ACM CCS, Virginia, USA, Oct. 2007,pp. 598–609.

[11] B. Wang, B. Li, and H. Li, "Public Auditing for Shared Data with Efficient User Revocation in the Cloud," Proc. IEEE INFOCOM, pp. 2904-2912, 2013.

[12] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.

[13] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores,"Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07),pp. 598-610, 2007.

[14] H. Shacham and B. Waters, "Compact Proofs of Retrievability,"Proc. 14th Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT'08), pp. 90- 107, 2008.

[15] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring Data Storage Security in Cloud Computing," Proc. 17th ACM/IEEE Int'l Workshop Quality of Service (IWQoS'09), pp. 1-9, 2009.

[16] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling Public Verifiability and Data Dynamic for Storage Security in Cloud Computing," Proc. 14th European Conf. Research in Computer Security (ESORICS'09), pp. 355-370, 2009.

[17] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," Proc. IEEE INFOCOM, pp. 525-533, 2010.

[18] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S.S. Yau, "Dynamic Audit Services for Integrity Verification of Outsourced Storages in Clouds," Proc. ACM Symp. Applied Computing (SAC'11), pp. 1550-1557, 2011.

[19] C. Wang, Q. Wang, K. Ren, and W. Lou, "Towards Secure andDependable Storage Services in Cloud Computing," IEEE Trans. Services Computing, vol. 5, no. 2, pp. 220-232, Jan. 2012.

[20] Y. Zhu, G.-J. Ahn, H. Hu, S.S. Yau, H.G. An, and C.-J. Hu, "Dynamic Audit Services for Outsourced Storages in Clouds,"IEEE Trans. Services Computing, vol. 6, no. 2, pp. 227-238, Apr.-June 2013.

[21] eXo Cloud IDE. (2002)Online code editor. Cloud IDE. [Online]. Available: https://codenvy.com/

[22] B. Wang, B. Li, and H. Li, "Oruta: Privacy-preserving public auditing for shared data in the cloud," in Proc. of IEEE CLOUD 2012, Hawaii, USA, Jun. 2012, pp. 295–302.

[23] B. Wang, L. Baochun, and L. Hui, "Public auditing for shared data with efficient user revocation in the cloud," in Proc. Of IEEE INFOCOM 2013, Turin, Italy, Apr. 2013, pp. 2904–2912.

[24] J. Yuan and S. Yu, "Efficient public integrity checking for cloud data sharing with multi-user modification," in Proc. of IEEE INFOCOM 2014, Toronto, Canada, Apr. 2014, pp. 2121–2129.

[25] D. Catalano and D. Fiore, "Vector commitments and their applications," in Public-Key Cryptography - PKC 2013, Nara, Japan, Mar. 2013, pp. 55–72.