Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 6.017

IJCSMC, Vol. 6, Issue. 3, March 2017, pg.102 – 109

REMOTE DATA INTEGRITY CHECK AND DATA UPLOADING USING CLIENT AND SERVER PROXIES

Mr. Abhinav N D[1], Ms. Arshiya Mubeen[2], Ms. Priyanka H V[3]

[1]Computer Science & Engineering, VTU, India
[2]Computer Science & Engineering, VTU, India
[3]Computer Science & Engineering, VTU, India
[1] ndabhinav93@gmail.com; [2] arshiyamubs@gmail.com; [3] preethihvt@gmail.com

Abstract: ***Large number of clients like to store data onto public cloud server (PCS) due to increase in advancement in cloud computing. As a result the new security problems are in need to be solved to help large number of clients in processing their data on public cloud servers. When the clients are not allowed to access the public cloud servers, they will be forwarded to proxy servers to process their data. Along with that, checking data integrity in that remote place is also a major security issue in public cloud storage. In this, it helps the clients in sending their data to the server via proxy and downloading the whole data with security. From the intension of solving security problem, we propose a proxy server oriented data uploading and remote integrity check of the data based on identity. For the same public key encryption and decryption is used. The remote data integrity check using proxy server with partial data method is used to address the problem. Our algorithm is efficient and very flexible. Based upon the real clients authorization, our protocol can realize private data integrity check using partial data.***

Keywords— Cloud computing, identity oriented – cryptography, proxy based public key cryptography, remote data integrity check, PCS, Original Client

I. INTRODUCTION

Cloud computing has been a latest trend in now a days. Different types of services are been provided from different type of cloud service providers. Huge and large amount of data are been stored on the servers present in remote places. The users of the cloud are also increasing now a days. The various types of services provided by different cloud service providers are massive storage for the different types of the data, tools for managing and processing different types of data. All these are possible because of cloud been made a public platform. Any users from any part of the world can store the data, fetch the data processes the data, manipulate the data and many more.

Although cloud storages have huge advantages, some challenges with security issue are to be encountered for cloud storage needs to be accepted by all the users. The cloud servers store many data of different users will prefer attack target and the data's are being facing a wide range of warning and attacks. Even more, different from usual type of data storage process, in cloud the owners of the data need not possess data physically after data is sent out onto the cloud service provider (CSP) who are not trust worthy. For once own advantages, cloud service providers might ignore a part of less frequently accessed data to save storage space. And, cloud service providers may be forced to hide the data corruptions caused by cloud server hackers to maintain reputation. It has been recognized that the security issues such as data integrity and availability are the main hurdles for the storage of data on to the cloud to be successfully adopted.

As this is going on increasing, the security issues and concerns for the same is also been increasing day by day. Providing confidentiality, integrity, security and availability of data are also been increasing daily. Since, users are storing their data on the public cloud platform and performing all the processing from server itself, providing confidentiality ,integrity, security and availability of data on the public cloud platforms are also been increasing daily. Users are expecting security for their data in various aspects. For the same we provide remote Data Integrity Check using Proxy Server with Partial Data method is used to address the problem. Our algorithm is efficient and very flexible. Based upon the real clients authorization, our protocol can realize private data integrity check using partial data**.**

II. MOTIVATION

The public cloud space is an environment for most clients for uploading their data to public cloud servers and databases and check their uploaded data's remote integrity check by using internet. Client, when takes the part of an individual manager, many real time problems are likely to occur. In case, the manager is playing the key role in the commercial frauds, he will be caught by the police authority. During the time of the investigation and interrogation, the manager will not be allowed to access the network and will restricted from accessing network in order to protect from collusion. But manager will be given legal authorization to carry out business at the complete duration of investigation.

When huge amount of data is generated, whom will help the manager from processing these data? If this data is not allowed to be process just in required time, he the responsible person will face the loose of economical interest. In order to avoid this from taking place, the manager will have to allow the proxy in processing it's data, for example, his secretary means subordinate. But, manager will no way think of others having the talent in performing the remote data integrity check using partial data. Public verification can lead to some endanger of giving up privacy. As an example, all the huge data that is stored may be identified from the unauthorized malicious verifying persons. When this uploading of the huge data amounts is confidential, private and remote integrity check of the data becomes very must and should in all the accepts. Even though the subordinate has the capacity of processing and uploading the data on behalf of the manager, he is still not in the mood to verify manager's remote integrity check of the data until he is representative for the manager. We call the subordinate, the proxy or alias of the manager.

In public key infrastructure, remote integrity data check protocol will take care of the certificate managements. When the head represents some of entities for performing the remote data integrity checking, it will incur considerably delays as the verifying officer will verify the certificate when he verifies the remote data integrity. In public key infrastructure, the acceptable delays usually come from the huge certificate verifications, and certificates creation, delivering, revocations, renewals, etc. At public cloud, computing end devices may be having less computational capacity, such as smart phones, ipod, etc. Based on the identity, public key cryptography is able to remove all the complex certificates management. In order to enhance the potentials, remote data integrity check using proxy server with partial data is more attractive. Hence, it is mandatory to study the ID-PUIC protocol.

III. CONTRIBUTION

On a public cloud platform, users are usually large in number. In this paper our major focuses is on remote data integrity check using proxy server with partial data. By using identity based public key cryptography, our proposed ID-PUIC protocol is more effective because the certificate management is completely terminated. ID-PUIC is a novel remote data integrity check using proxy server with partial data model in public cloud server. We will display the system architecture and security model for ID-PUIC protocol. Thus, based upon the bilinear pairing method, the design of the first concrete ID-PUIC protocol is based. According to the random oracle model, our proposed ID-PUIC protocol is provably secure.

On the basic of original client's authorization, our protocol can effectively realize private data checking, and public checking.

IV. RELATED WORK

There are many variety of security problems and issues in the cloud computing [1], [2]. Our paper is based upon various research results on proxy based cryptography, identity oriented public key cryptography and remote integrity check of the data on public cloud servers. In most of the sceneries, the cryptography operation is been represented to by the third party, for example proxy. Hence, we are bound to the proxy based cryptography. Proxy based cryptography is an important cryptography primitive unit. During 1996, *Mambo et al.* expressed the notions on the proxy based cryptosystem [3]. With the help of bilinear pairings been brought into the identity based cryptography, identity based cryptography has become the most effective and practical.

As the identity based cryptography has become more effective due to the property that it avoid the certificate management, large and large experts are suited to study identity 0based proxy cryptography. At 2013, *Yoon et al.* presented with an ID oriented proxy signature system and scheme with message recovery [4]. *Chen et al.* proposed an proxy signature idea and a brink proxy signature format from the Weil coupling [5]. By combining the substitute cryptography with encryption procedure, some proxy re00encryption plan are anticipated. *Liu et al.* sanctify and constructed the feature00based proxy signature [6]. *Guo et al.* presented a noninteractive CPA (selected plaintext attack)-secure proxy reencryption idea, which is opposed to collusion attack in forge re-encryption keys [7]. Many other concrete proxy re-encryption schemes and their applications are also proposed [8]–[10].

At the public cloud servers, remote data integrity checking is a important security issue. As the clients immense data is out of their organize, may clients data may be infected by the malevolent cloud servers despite the consequences of deliberately or not deliberately. In order to address the original safety problem, more efficient model is offered. In 2007, *Ateniese et al.* proposed attestable data possession (ADP) paradigm [11]. In ADP model, the checking can verify the remote data integrity without retrieval or download of the complete data. ADP is a probabilistic evidence on remote data integrity check by inputting random set of blocks from the public clouds servers, which significantly reduces I/O costs. The examiner can carry out the remote data integrity checking by maintaining small data about the data.

Following that, some dynamic PDP models and rules are considered [1]–[6]. Following *Ateniese et al.*'s revolutionary work, many remote data integrity scrutiny models and protocols have been projected [7]–[9]. In 2008, proof of retrievability (POR) method was put forth by *Shacham et al.* [2]. POR is a stronger model which makes the overseer not only check the remote data reliability but also fetch the remote data. More POR proposal have been proposed [1] to [6]. On some crate, the client may assign the remote data integrity checking undertaking to the third party.

In cloud computing, the third party audit is vital [2],[3]. By making use of cloud storage services, the customers can access the remote data with autonomous geographical places and areas. The end devices may be portable and limited in totalling and storage. Hence, effective and security based ID-PUIC protocol is more appropriate for cloud clients capable of using as mobile end devices.

From the role of the remote data integrity inspector, all the remote data integrity examination protocols are classified into two categories: private remote integrity data check and public remote data integrity check. In the retort inspection phase of private remote data integrity checking, some private information is central. On the counter part, secret information is not required in the comeback checking of public remote data integrity check. Particularly, when the private information is handed over to the third party, the third party can even execute the remote data integrity checking. In this case, it is also called delegated checking.

Ateniese et al. [6] was the first to pioneer the “Provable Data Possession (PDP)” mould and projected an integrity substantiation scheme for standing data using *Rivest, adi shamir,* based homomorphic authenticator. During the same period, *Juels et al.* [8] proposed the “Proof of Irretrievability (PoR)” model which has more strength than the PDP model in the wisdom that the system in addition guarantee the retrievability of outsourced data.

Data truthfulness proof in cloud storage by *Sravan Kumar* provides a plan for static storage of data [2] with uncovered minimum cost and less endeavour. To certify confidentiality, integrity and authentication of the actual data, a reliable service based on faithful encryption plot is provided with unyielding access pedals and planned data backup.

In particular, the proposers proposed a on spot verification approach to undertaking ownership of data records and engaged error correcting coding technology to ensure the retrievability. As a constraint on their scheme is that the number of challenge is inhibited. *Shacham et al.* [10] utilized the homomorphism signature in [2] to design an improved PoR scheme.

V. PROPOSED SYSTEM

As we have seen that on the public cloud, the providers of the cloud service must take care of the security problems. In account of the same, here we put forth an architecture or a system model and the protocol associated with it on which it works. Our model is effective and efficient in providing authentication, authorization during the access of the data and also ensures the integrity of the data stored on the public cloud.

Our system model provides the security for data stored by the people on the cloud by allowing to access right data by the right clients. This is the security provided at the client side. Also, secure uploading of the data is provided.

Once the data is been uploaded we don’t know the exact geographical location, where the client data is stored. Hence, we need to provide integrity for the data stored at remote places. Our method called remote data integrity check using partial data provides security for the clients data during data uploading and provides security for the data stored in remote place by integrity checking of the data stored in remote place with the partial data. Here we are using ID—PUIC protocol to accomplish our system model and provide security. This is one of the efficient protocol among the other protocols available to address the same security issues.

Architecture:

Here we are proposing an protocol on the basics of which our paper working stands. The protocol is called as ID—PUIC protocol. The protocol consists of four different type of entities or components. In order to provide the cryptography function which provides security for the data uploading, us are using Gap Diffie

Hellman key exchange. The various components can be called as individual modules at the time of execution of the components in the system.

The components are:

1. Actual / Original client (O).
2. Public cloud server (PCS).
3. Client Side Proxy.
4. Server Side Proxy.
5. PKG (Private key Generation), isn general can be called as key distribution center (KGC).

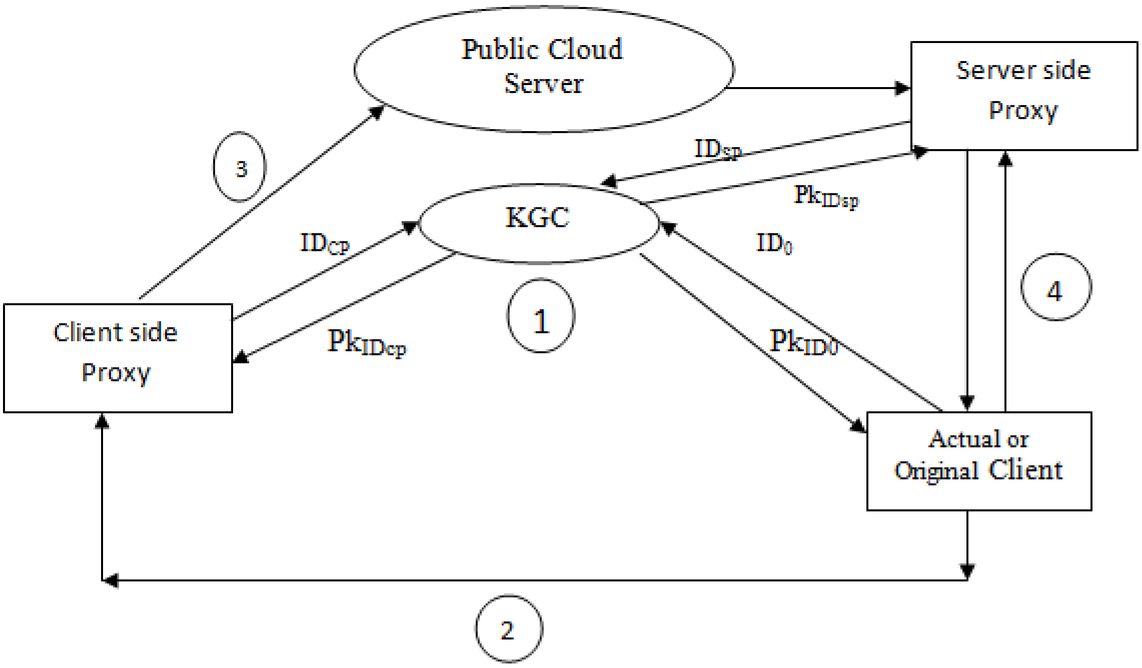

Fig 2: Internal Architecture of ID-PUIC

Actual Client:

Client is an component which has huge amount of data to be uploaded to the public cloud server. Uploading is been done by proxy which can function the remote data integrity check.

Public cloud server (PCS):

This is the entity which is provided by the public cloud service provider having magnificent space for storing of clients data. It is also providing the resources for performing computations on the data that is stored on cloud.

Client Side Proxy:

The authorized components for processing the original or actual clients data and upload them, is chosen and authorized by actual client. When the proxy is satisfied by the warrant which is generated and signed by actual client, it can process data and upload the original or the actual clients data; else proxy cannot perform this action.

Server Side Proxy:

Server side proxy is the unit responsible for providing integrity check. Client can perform his data integrity via server side proxy only as it is the authorized person for performing the check. When the proxy is satisfied with the warrant which is generated and signed by the actual client, it can process the integrity of the clients data, otherwise, server side proxy cannot perform the action.

Key generation center (KGC):

This is the component after receiving / accepting or inputting the identity, generates the private key corresponding to the accepted identity. In our protocol, actual client will interact with the public cloud server to perform remote data integrity checking. Our ID—PUIC protocol comprises of four phases and an interactive system. The four phases and an interactive system are detailed below :

Setup Phase:

This is the first phase in our protocol. It takes a security parameter “s” as input, and an system public parameter and a master secret key are given as an output. Out of these, public parameter is made as public and the other parameter is made secret i.e. master secret key (msk). Is kept secret by the key generation center.

Extract:

After the setup phase , when the system public parameter , master secret key and an ID are given as inputs , key generation center outputs a unique private key (pk_{ID}) corresponding to the identity of the devices. The devices here are proxy and actual client. Client’s can be of any in number.

Proxy-key generation:

This is the third phase where the original client creates a warrant and signs the warrant “w” . After this, client sends the warrant – signature pairs to the proxy. On receiving the same, from the client, the proxy creates another key at its end called as proxy key with the help of its own private key.

Tag Generation:

Actual client gives a block of the data or file F_i and a proxy key as input to the proxy. Then, the proxy generates a Tag T_i corresponding to each block. This block – tag pair is then sent to public cloud server. And, block – tag pair is unique for each of the data that the actual client sends to the cloud.

Interactive Proof:

This is an interactive system which provides an communication chance between public cloud server and original client. After the end of the session, actual client outputs { 0,1} denoting “0” for failure and “1” for success.

Block Diagram:

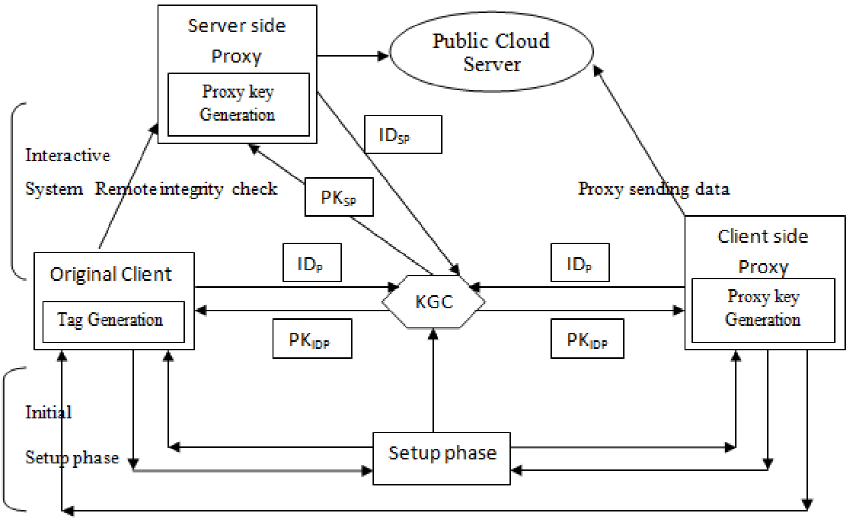

Fig 2 : Internal Block diagram of the IDPUIC

Now let us see our block diagram of our proposed system. First, setup is done and all the system necessary parameters are generated . Based on these system generated parameters, other processes are been performed.

1. This phase is called "Extract phase" , where the identity of the entity is given as input and the key generation centre generates private key of the corresponding entity. Specially, it generates the private key of the corresponding client and proxy.
2. In this phase, proxy entity generates the proxy key, the original client generates its warrant and extents help for the proxy in the generation of the proxy key. Client side proxy performs secure upload of the client data and the server side proxy perform the integrity of the client data. The integrity of the data is carried out by the client via server side proxy as the intermediate.
3. Tag generation phase is an important phase. When the clients data or file is as input , the proxy generates a corresponding "Tag" for the block of data and will upload the block –tag pair onto the public cloud server.
4. Interactive system: In this phase the actual client will interact with the public cloud server and via the interactions , the original client checks it's data 's integrity as its data is stored in the remote geographical location which is unknown to the original client.

VI. ADVANTAGES OF THE PROPOSED SYSTEM

- Computation speed is very fast. The generation of the secret key and etc takes place at a faster rate.
- As we are uploading files of data, data blocks of any size "n" can be uploaded and its integrity can be checked at a very faster rate.
- The same can be implemented using c programming language.
- As we have applied to public cloud server , similarly we can apply for the hybrid cloud . In this scenario proxy can be made to act as the private cloud for the original client.
- The extract, tag generation are the important security phases which provides security for the data that is been uploaded to the cloud i.e. public cloud.
- The concrete ID-PUIC protocol is provably secure and efficient for using the formal security proofs and effective analysis.

VII. CONCLUSION

Our proposed ID-PUIC protocol provides an efficient and an effective method for remote data integrity check and upload of data onto the cloud. The extract phase, Tag generation phase, key generation centre, interactive system play an important role in providing the security . Hence, the security issues regarding remote data upload and integrity checking of the remotely uploaded data is been achieved successfully.

VIII. FUTURE WORK

In our discussion, we went up with too many phases like setup, tag generation etc. It becomes too congested from the point of view of the architecture. The same can be solved using automatic key generation algorithms or virtual key generation systems.

REFERENCES

[1] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing," *IEICE Trans. Commun.*, vol. E98-B, no. 1, pp. 190–200, 2015.

[2] Y. Ren, J. Shen, J. Wang, J. Han, and S. Lee, "Mutual verifiable provable data auditing in public cloud storage," *J. Internet Technol.*, vol. 16, no. 2, pp. 317–323, 2015.

[3] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in *Proc. CCS*, 1996, pp. 48–57.

[4] E.-J. Yoon, Y. Choi, and C. Kim, "New ID-based proxy signature scheme with message recovery," in *Grid and Pervasive Computing* (Lecture Notes in Computer Science), vol. 7861. Berlin, Germany: Springer- Verlag, 2013, pp. 945–951.

[5] B.-C. Chen and H.-T. Yeh, "Secure proxy signature schemes from the weil pairing," *J. Supercomput.*, vol. 65, no. 2, pp. 496–506, 2013.

[6] K. Huang, J. Liu, M. Xian, H. Wang, and S. Fu, "Enabling dynamic proof of retrievability in regenerating-coding-based cloud storage," in *Proc. IEEE ICC*, Jun. 2014, pp. 712–717.

[7] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[8] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 5, pp. 847–859, May 2011.

[9] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," *IEEE Trans. Services Comput.*, vol. 5, no. 2, pp. 220–232, Apr./Jun. 2012.

[10] 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 Comput.*, vol. 6, no. 2, pp. 227–238, Apr./Jun. 2013.