



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 5, Issue 5, May 2017

An Innovative Pliable Distributed Cloud Storage Integrity Checking Scheme

Ulya Sabeel

Assistant Professor, Dept. of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University, Haryana, India

ABSTRACT: Cloud computing has been a widely proliferous technology providing promising services to the users in the recent years and is ratified by most of the organizations. Archiving colossal amounts of data in the local storage is too pricey and has high administration costs. Cloud computing is a location independent service wherein data can be accessed anytime anywhere and is contracted out as a service by the users to the third-party cloud server. Such data may have sensitive profitable information that needs to be secured. Several schemes have been proposed to handle the attestation of remote data integrity in the cloud archive systems. Most of these schemes, however, do not bolster adept data dynamics and suffer from security loopholes under these conditions. In this paper, a secure and innovative scheme of ensuring a pliable distributed cloud storage by ensuring the integrity of the data has been proposed. It insures the protected archiving of our private data in cloud storage environment in a proficient manner which requires less time and low computational power thereby censoring the intruder from infiltrating into our private storage. The scheme is straightforward and uncomplicated as compared to the others. The prototype framework has been developed in C#. The experimental results proclaim that the framework is effective and pliable for distributed cloud storage integrity checking.

KEYWORDS: Cloud Repository, Data Integrity, Pliable, Security, Third Party Auditor (TPA)

I. INTRODUCTION

A multitudinous number of businesses are contracting out their enormous amounts of data to the Cloud based repositories, apparently by paying a charge for the privilege and storing their crucial data dutifully and accessing it whenever required. This method provides great assistance to the users owing to its simplicity in the areas of hardware and software management, storage management, power management global data access, personnel management and cost management. It can also insure a decent repertory of crucial information by preserving its numerous replicas thereby minimizing the chance of hardware failures causing data loss [1].

The advantages of cloud based repositories are compelling but due to the opaque nature of the cloud many security issues exist. These issues need to be thoroughly probed for providing a reliable data solution. Data security, privacy, integrity and trust are the major impediments faced by cloud companies for their global acceptance [2]. To curb these issues, many algorithms and protocols (MD5, RSA, PDP) have been designed and implemented [3], [4].

Several data leakage cases have been reported for the most eminent storage service providers like Google cloud, Dropbox and Amazon S3 [6], [7], [8] and [9]. The data loss might be concealed by the cloud service provider to maintain its stature or data might be dispensed to save repository space, while affirming the data diminution to be nil. The existing security mechanisms claim that the cloud supervisor can't be fraudulent which is not true in all the cases. Therefore, reconstruction of an adept data integrity algorithm for cloud repositories should receive prime attention [5].

In this paper, an innovative model for ensuring a pliable distributed cloud storage by maintaining the integrity of its contents has been proposed. This verifies that the data saved by the user in the cloud repository has not been compromised, thereby assuring data integrity. This scheme will prevent the adversaries from misstating or falsifying



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 5, Issue 5, May 2017

any information stored by the user by frequent inspections on such cloud repositories. It allows the data owner to efficiently and securely guarantee the authenticity and reliability of its data provided by the cloud service provider. The data owner might be a small hand-held device like a mobile phone or a tablet with limited processing capabilities like CPU power, communication bandwidth and battery. Our proposed method will be able to address these issues and generate a proof without involvement of the server or client to access the entire data. This scheme also mitigates the local client computation and bandwidth consumption.

II. RELATED WORK

The proof for data integrity in the cloud storage environment has gathered a lot of research attention. For assurance of remote data integrity in the unreliable cloud, many scrutinizing techniques have been proposed. As the amount of data generated is exceeding the data storage limit, it is expensive for small organizations to constantly modernize their hardware and maintain large storages whenever supplementary data is generated. This complication is further aggravated with consumption of heavy bandwidths for large file transfers with the system having only limited CPU and battery power [1].

In 2004, Boneh et.al proposed [11] single keyword searchable encryption where all users can send the data using public keys but only authentic users can search through the data using their allotted private keys. However, the encryption technique used enhances the complexity of this method. In [12], a Provable Data Possession Scheme has been proposed by the authors. This scheme ensures whether the data stored in the Cloud repositories is fully retained by the remote cloud server by generating metadata and comparing the hash values. This scheme has high overhead and consumes more time because the hash is run over the entire file. In 2007, Ari Juels et.al proposed [10] a scheme by using sentinels for Proof of irretrievability for large files. It used a single key irrespective of the file size or number that need to be verified. Whilst this scheme consumes less time, it cannot handle dynamic data and increased number of queries. Also, this involves encryption of file using a secret key, which is more complicated to handle especially when the file is very large. In [13], the authors have discussed compact proofs of retrievability using two POR schemes based on the homomorphic linear authenticators for private and public verification. Although the speed of verification is enhanced here, but the processing costs too are high. Dodis et.al. in his scheme [14] reduced the size of the message, but the scheme still suffered from the limitation due to linearity between the length of proof response to the number of elements in the data block. Also, this scheme could only support private verification of data resulting in increased overhead on the data owner. In [17], the authors have proposed a privacy preserving authorization system for the cloud which processes the microdata and sends anonymous data to the cloud service provider for integration with additional information to get the results. In [16], the authors talk about Privacy-Aware Data Storage and Processing in Cloud Computing Architectures. This paper deals with tamper proof cryptographic coprocessor, configured by authentic third party to provide a secure platform free from unauthorized access. The authors in [15] have discussed a case study "RACS" for Cloud Storage Diversity to avoid vendor lock-in and essentially minimize the costs.

In [18], the authors, proposed a secure auditing scheme, but suffered from the limitations of high computational costs which were directly proportional to group and data size. Wang et al, in [19] proposed a scheme for public auditability and data dynamics for storage security in cloud computing. However, this scheme also suffers from high computational costs linear to the size of the data. In [20], Ming Li et.al proposed an Authorized Private Keyword Search over Encrypted Data in Cloud Computing. Here multiple data owners encrypt their documents and use keywords and indices to allow searches. This scheme also supports multi-dimensional range queries however, suffers from the limitations of high computational costs and excessive overhead. Ning Cao et.al in [21] proposed privacy preserving multi-keyword search where a user can search the cloud data with multiple query keywords. This scheme also suffers from the limitations of high computational costs and assumes that the cloud server can be trusted every time. In reference [22], the authors propose a method of providing a choice of encryption algorithms to the users, who can select any algorithm to secure their data. The authors here have not considered the cost of maintaining all these algorithms. Also, this method can only be used for static data and limited number of queries. In [23], authors talk about 3 level security for the user, but they have not considered the cost aspect of this model. Also, it suffers from the challenges such as Data locks by cloud provider, fault tolerance and disaster recovery mechanisms. In [24], the authors



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

have proposed a fuzzy identity-based data auditing mechanism, where a user's identity can be viewed as a set of descriptive attributes, however, authors have not considered the cost and complexity aspect of this prototype. Also, they have not tested it in a real-time environment. In [25], Salah H. Abbdal et. al, have proposed a mechanism to verify the data integrity using TPAs based on homomorphic linear authentication and an elliptic curve digital signature algorithm to support public verifiability. Although this technique reduces the time complexity, they have not considered the cost and overhead involved. Yong Yu et al. in [26], have proposed ID based remote data integrity checking technique which uses key-homomorphic cryptographic primitive to reduce the system cost and complexity for a PKI Framework. Although, the security is enhanced, the high cost estimate is not taken into consideration. In [27], the authors have proposed a novel public verification scheme for the cloud storage using indistinguishability obfuscation. However, the authors have not discussed how to resist a malicious auditor and how to reduce the cloud server overhead.

Keeping in mind all these issues, a secure scheme of ensuring a pliable distributed cloud storage integrity scrutinizing maneuver has been proposed.

The rest of the paper is organized as follows: section III consists of proposed scheme, section IV describes the experimental results, and finally the conclusion is given in section V followed by references in section VI.

III. PROPOSED SCHEME

One of the most cardinal concerns about cloud storage is maintaining the integrity and correctness of data stored for the client. As the data is stored at a remote location, there should be a strategy for the client to scrutinize its integrity. In this paper, a pliable distributed cloud storage integrity scrutinizing maneuver has been proposed. The corroboration can be agreed upon by both the client and the cloud service provider in the form of a Service level agreement (SLA). This scheme is used to check whether the data has been illegitimately amended or deleted.

The proposed scheme involves encrypting only few bits per block, but not the entire data thus reducing the computational overhead. A high likelihood of security can be attained by encryption of fewer bits instead of the entire data. The client does not need to store any data within itself, thus minimizing the storage overhead and bandwidth requirements. Therefore, the scheme is well suited to thin clients (mobile phones, tablets, etc.) as well.

A. System Architecture:

Irrespective of the data size, the Third-Party Auditor (TPA) is required to store only one cryptographic key and two random sequence generating functions. The Auditor is not supposed to store any data with itself. Its job is to pre-process the file, append meta-data onto it and store it to the cloud repository. During the verification process, the Auditor uses the meta-data to verify the actual data integrity. This scheme only scrutinizes the correctness of the data, but cannot prevent data loss from cloud data repositories due to natural or unnatural causes. To prevent data loss, it should be duplicated and stored in different authentic data centers. To support dynamic data behavior and multi-query operations like update, insert and delete on the client side, an additional encryption of fewer data bits is done. The architectural diagram for this scheme is given in figure 1. The key features of the proposed scheme are as follows:

1. Simplicity of operation.
2. low communication cost as no complex encryption algorithms have been used
3. low computation cost for data authentication
4. low storage and bandwidth overhead for client.
5. reduced chance of data loss due to hardware failures
6. protected archiving of our private data

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 5, Issue 5, May 2017

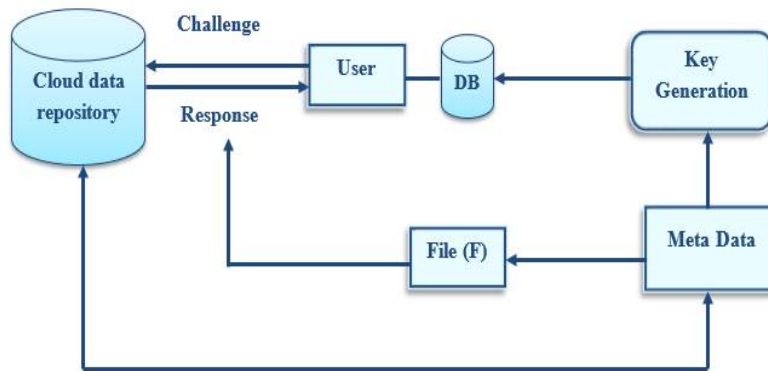


Figure 1: Architectural Diagram

B. Integrity Scrutinizing Maneuver Algorithm

The proposed technique consists of three entities: Owner (Client), TPA (Third Party Auditor) and the Admin. Each entity has a major role to play for the algorithm to execute perfectly. The client is the owner of the data who wants to use the cloud services. The role of the Auditor is to pre-process the data, add meta-data to it and store it in the cloud repository. The Admin is the main entity that controls the functioning of both the client as well as the TPA. The sequence diagram is shown in figure 2.

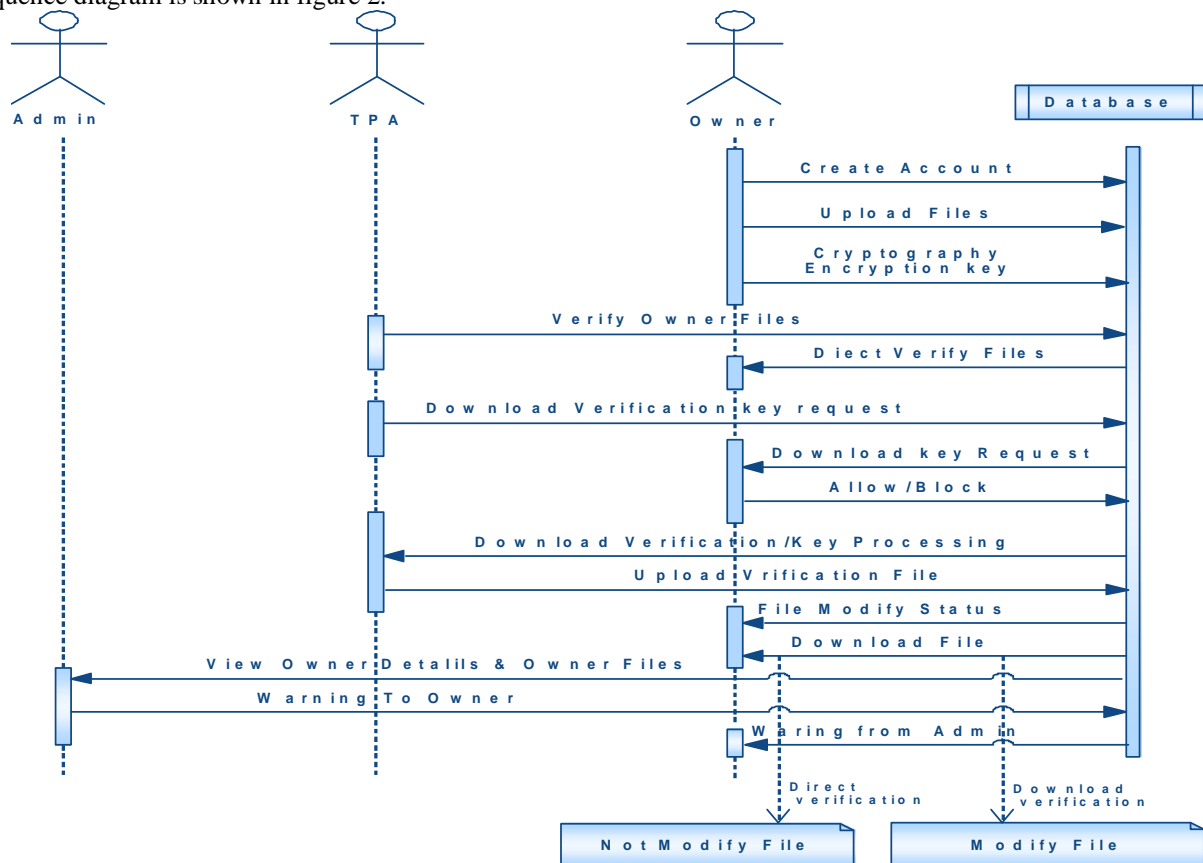


Figure 2: Sequence Diagram



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 5, Issue 5, May 2017

The proposed algorithm has two parts:

Algorithm 1: General

Step 1: The client (owner) of the data registers with the cloud services.

Step 2: A secret key for login is sent to the email id of the client for secure access of the cloud.

Step 3: After login, client uploads its file to the cloud repository. A cryptographic encryption key is sent to the mail id of the client.

Step 4: Every client file uploaded or downloaded to and from the cloud repository goes through the encryption scheme by generating new keys for each piece of data.

Step 5: Admin overviews the entire functioning, monitors client data and issues warnings in case of suspicious activities.

Step 6: The Auditor verifies the client files, appends metadata to the file and stores them in the repository.

Algorithm 2: Generation of Metadata and Integrity Checking

Step 1: The client wants to store the file (F) in the cloud repository. Each file is divided into ' i ' blocks.

Step 2: Each ' i ' block is divided into ' j ' bits.

Step 3: ' k ' number of bits out of ' j ' bits of ' i ' blocks are selected for the construction of Metadata.

Step 4: Generation of metadata is done by the function $H(m,n)$ which is elucidated as follows:

$$H(m, n) \rightarrow \{1..j\}, m \in \{1..i\}, n \in \{1..k\} \text{ ----- (1)}$$

Where ' k ' is the number of bits per block. Function $H(m,n)$ gives the n^{th} bit in the m^{th} data block. Value for ' k ' is a secret given by the Auditor. Each data block has ' k ' bits and total bits for all ' i ' blocks is given as $(i*k)$ bits. " jm " represents the k bits of meta data form m^{th} data block.

Step 5: The metadata from the data block " jm " is encrypted and modified to metadata " Jm ". Let " G " be the function which generates k bit integer " am " for each m . This is kept a secret with the Auditor and is defined as:

$$G: m \rightarrow am, am \in \{0..2^i\} \text{ ----- (2)}$$

Step 6: For metadata " jm " of each data block the number am is added to get a new k bit number given below:

$$Jm = jm + am \text{ ----- (3)}$$

Step 7: The metadata generated is now clubbed together and affixed to the user's file F before saving it to cloud.

Step 8: If the client wants to verify integrity of the m^{th} data block, the Auditor throws a to the cloud server by specifying the block number m and bit number n generated using the function " H ", which is known only by the Auditor. The Auditor also specifies the position at which the metadata corresponding to block m is appended. Metadata is a k bit number.

Step 9: The cloudarchive server sends the response for the verification to the client via Auditor.

Step 10: Metadata of the response is decrypted using " am ". The bit in decrypted metadata is compared with the bit send in the response by the cloud. If both match, then the data is intact with its integrity maintained and otherwise if a mismatch is found. In other words, the cryptographic key generated by TPA is compared to the cloud archive's cryptographic key. If both match, then the data integrity is not breached and vice versa.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 5, Issue 5, May 2017

IV. EXPERIMENTAL RESULTS

This section describes the environment, hardware and software requirements that have been used to design the proposed framework and to perform the experiments successfully. The system requirements are given in table I.

TABLE I. SYSTEM REQUIREMENTS

S.No.	Parameter	Value
1	System (PC)	Intel Core i3 processor
2	Operating System	Windows 10
3	Database	SQL Server 2008
4	Framework	Microsoft Visual Studio 2010
5	Coding language	C#

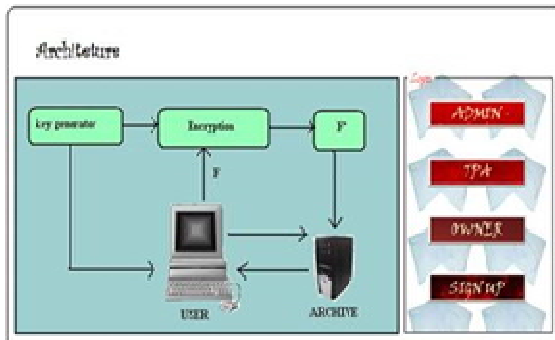


Figure 3: GUI for the proposed framework



Figure 4: Client/Owner Registration page

Figure 3 depicts the GUI for the proposed framework which is developed in C#. It has the following modules: Admin, TPA, Owner and Signup for the client. Figure 4 shows owner/client registration process. It consists of credentials to be filled by the client such as Owner ID, Password, Gender, Age, Phone, Email ID and Date.

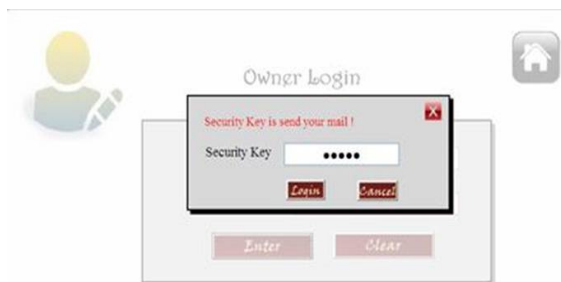


Figure 5: Client/Owner Login page

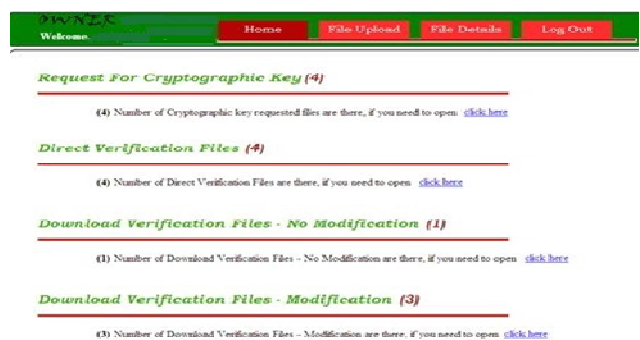


Figure 6: Client/Owner Profile

After the registration process, the client can login as shown in figure 5. The security key for the client is sent to his mail id. To enhance the security of the client account, the security key can be changed by the client later per his own

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

preference. Figure 6 depicts the account of the client/owner and all the tasks that he can perform like file upload, checking the file details, verification, modification and download of the files.



Figure 7: File upload status for the client



Figure 8: File verification status for the client.

Figure 7 portrays the file upload status and generates a file cryptography encryption key which is sent to the owner's mail id. Figure 8 illustrates the file verification process after the two keys have been matched and found to be same and the integrity of the data has been verified.

V. CONCLUSION

In this paper, an innovative prototype framework for ensuring a pliable distributed cloud storage by checking the data integrity has been proposed. The main aim of this research is to develop a pliable framework for checking the integrity of the data stored in a cloud repository by the client. The proposed framework removes the drawback of the existing techniques by providing features like simplicity, low communication and computation costs as it uses least complex encryption techniques, low storage and bandwidth requirements for the client, reduced chance of data loss due to hardware failures and protected archiving of our private data. The client is needed to store only two functions, the bit generator function H , and the function G which is used for encrypting the data. Therefore, the storage at client side is minimal, thus it can be used for thin clients with low power batteries. Additionally, the encryption process used in our technique is not so complex. Only a fraction of client data is encrypted, thus reducing the time complexity of this framework. The network bandwidth is minimized too because the cloud repository sends the results in fewer bits as compared to other techniques. In future, the focus of this research will be on using data in a dynamic and multi-query environment.

REFERENCES

- [1] Sravan Kumar R, Ashutosh Saxena, "Data Integrity Proofs in Cloud Storage", Third International Conference on Communication Systems and Networks (COMSNETS 2011), IEEE 2011
- [2] A. Shawish and M. Salama, Cloud Computing: Paradigms and Technologies, F. Xhafa and N. Bessis (eds.), Intercooperative Collective Intelligence: Techniques and Applications, Studies in Computational Intelligence 495, DOI: 10.1007/978-3-642-35016-0_2, Springer-Verlag Berlin Heidelberg 2014
- [3] S. Subashini and V. Kavitha, A survey on security issues in service delivery models of cloud computing, Journal of Network and Computer Applications, 34(1): p. 1-11, 2011.
- [4] C. Ning., et al. Privacy-preserving multi-keyword ranked search over encrypted cloud data, INFOCOM, 2011 Proceedings IEEE. 2011.
- [5] Y. Zhang et al., "SCLPV: Secure Certificateless Public Verification for Cloud-Based Cyber-Physical- Social Systems Against Malicious Auditors," *IEEE Trans. Computational Social Systems*, vol. 2, no. 4, pp. 159-170, 2015.
- [6] Amazon forum. major outage for amazon s3 and ec2, <https://forums.aws.amazon.com/thread.jspa?threadID=19714&start=15&tstart=0>.
- [7] Amazon web service. summary of the amazon ec2 and amazon rds service disruption in the us east region, <http://aws.amazon.com/message/65648/>.
- [8] Business insider. amazons cloud crash disaster permanently destroyed many customers data, <http://www.businessinsider.com/amazon-lost-data-2011-4>.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 5, Issue 5, May 2017

- [9] Dropbox. dropbox forums on data loss topic, <http://forums.dropbox.com/tags.php?tag=data-loss>.
- [10] A. Juels and B. S. Kaliski, Jr., "Pors: proofs of retrievability for large files," in CCS '07: Proceedings of the 14th ACM conference on Computer and communications security. New York, NY, USA: ACM, pp. 584-597, 2007.
- [11] D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public Key Encryption with Keyword Search," Proc. Int'l Conf. Theory and applications of Cryptographic Techniques (EUROCRYPT), 2004.
- [12] Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song. "Provable Data Possession at Untrusted Stores." ACM Conf. Computer and Comm. Security (CCS '07), 598-609, 2007."
- [13] H. Shacham and B. Waters. "Compact proofs of retrievability", ASIACRYPT'08, Berlin, Heidelberg. Springer-Verlag, pages 90-107, May 2008.
- [14] Y. Dodis, S. Vadhan, and D. Wichs. Proofs of retrievability via hardness amplification. TCC'09, Berlin, Heidelberg, pages 109-127, 2009.
- [15] A.Libdeh, L. Princehouse, and H. Weatherspoon, RACS: A Case for Cloud Storage Diversity, SoCC 10:Proc. 1st First ACM Symposium on Cloud Computing , PP 209-240, 2010.
- [16] W. Itani, A. Kayssi and A. Chehab, Privacy as a Service: Privacy-Aware Data Storage and Processing in Cloud Computing Architectures, Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2009
- [17] D W. Chadwick and K. Fatema, A privacy preserving authorization system for the cloud, Journal of Computer and System Sciences ,PP 1359-1373, 2012.
- [18] G. Timothy and M. M. Peter. The nist definition of cloud computing. NIST SP: 800-145, September 2011.
- [19] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, Enabling public auditability and data dynamics for storage security in cloud computing, Parallel and Distributed Systems, IEEE Transactions on, vol. 22, no. 5, pp. 847-859, 2011.
- [20] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized Private Keyword Search over Encrypted Data in Cloud Computing," Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '10), pp. 383-392, June 2011.
- [21] Ning Cao, Cong Wang, Ming Li, Kui Ren, and WenjingLou, "Privacy-Preserving Multi-keyword Ranked Search over Encrypted Cloud Data" , IEEE Transactions on Parallel and Distributed Systems, Volume 25 Issue 1, January 2014
- [22] Kanmani P, Anusha S, "A Novel Integrity Scheme for Secure Cloud Storage", IEEE Sponsored 9th International Conference on Intelligent Systems and Control (ISCO), 2015
- [23] PreetiSirohi, Amit Agarwal, "Cloud Computing Data Storage Security Framework relating to Data Integrity, Privacy and Trust", First International Conference on Next Generation Computing Techniques (NGCT-2015), Dehradun, India, 4-5 September 2015.
- [24] Yannan Li, Yong Yu, Geyong Min, Willy Susilo, Jianbing Ni and Kim-Kwang Raymond Choo, "Fuzzy Identity-Based Data Integrity Auditing for Reliable Cloud Storage Systems", IEEE Transactions on Dependable and Secure Computing, Volume: PP, Issue: 99 ,01 February 2017
- [25] Salah H. Abbdal, Hai Jin, Ali A. Yassin, Zaid Ameen Abduljabbar, Mohammed Abdulridha Hussain, Zaid AlaaHussien, Deqing Zou, "An Efficient Public Verifiability and Data Integrity Using Multiple TPAs in Cloud Data Storage", IEEE 2nd International Conference on Big Data Security on Cloud, IEEE International Conference on High Performance and Smart Computing, IEEE International Conference on Intelligent Data and Security, 2016
- [26] Yong Yu, Man Ho Au, Giuseppe Ateniese, Xinyi Huang, Willy Susilo, Yuanshun Dai, and Geyong Min, "Identity-Based Remote Data Integrity Checking With Perfect Data Privacy Preserving for Cloud Storage", IEEE Transactions on Information Forensics and Security, Vol. 12, No. 4, April 2017
- [27] Yuan Zhang, Chunxiang Xu, Xiaohui Liang, Hongwei Li, Yi Mu, and Xiaojun Zhang, "Efficient Public Verification of Data Integrity for Cloud Storage Systems from Indistinguishability Obfuscation", IEEE Transactions on Information Forensics and Security, Vol. 12, No. 3, March 2017

BIOGRAPHY

Ms. Ulya Sabeelis is an Assistant Professor in the Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Haryana, India. She received Masters of Technology in Computer Science and Engineering degree in 2013 from Amity University Noida, India. Her research interests are Computer Networks (wireless Networks), Network Security, Mobile and Ubiquitous Computing, Cloud security and Green Computing.