International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 04 Issue 13
October 2017

Reliable cloud storage meetup with preserved network system

N. Srinivasa Varma
M.Tech,
Department of Computer Science and Technology,
SRKR Engineering College, Bhimavaram,
West Godavari, Andhra Pradesh, India.

K. Sravani
Assistant Professor,
Department of Computer Science and Technology,
SRKR Engineering College, Bhimavaram,
West Godavari, Andhra Pradesh, India.

Abstract

Cloud In cloud storage environment, data users host their data on cloud servers and users can access the data from cloud servers. Due to the data outsourcing, this process of data hosting service introduces new security challenges, which requires an independent auditing service to check the data integrity in the cloud. Because, data user need to be convinced that the data are correctly stored in the cloud. Storage auditing system could not be guaranteed to provide proper auditing result thus Third-Party auditing is the better choice for the storage auditing in cloud computing. A Third-Party auditor is capable to do a more efficient work and convinces both the cloud service providers and the data user. There are chances of data being lost or get misplaced in cloud storage environment. Recent studies have shown that hashing methods are effective for high dimensional nearest neighbor search. A common problem shared by many existing hashing methods is that in order to achieve a satisfied performance, a large number of hash tables (i.e., long code words) are required. To address this challenge, in this paper we propose a novel approach called Compressed Hashing by exploring the techniques of Secure Network Coding and compressed sensing.

Index Term:- Auditing, Secure Network Coding,Homomorphic Scheme, Compressed Hashing;

1. Introduction

Cloud computing is a kind of area where users can able to store data and can be accessed easily. Cloud server will manage the data that is to be stored under the user account. This helps to manage users data safely and overcome the traditional drawback of storing the data in systems hardware. Cloud is being used not only to store data but also as an inexpensive, efficient, and flexible to access. By using cloud we can do work on the data at any time at any place. Cloud server provides a chance to store our data in server through internet instead of storing at our local server. As we normally store our data in the cloud server there may be a chance of data loss which is difficult to recover. So, in order to solve this problem there is a need of secure cloud which will handle these kinds of problems. Based on this problem one of the researchers introduced the concept called Network coding in order to improve network

capacity, but it leads to pollution attacks. It is a powerful attack which can succeed even when the number of normal nodes is more than the lower bound. In this attack the attacked node is mislead to a special location, which leads to confusion of compromised node with normal node. It is also adds malicious packets while travelling in the networks. To overcome this, we implement the Homomorphic scheme; this technique provides data security while data is passing through nodes.

We propose two enhanced secure cloud storage protocols that can satisfy the needs of different applications and achieve the cooperation of different network entities. The basic idea is that each codeword in the network needs to be authenticated by checking if the codeword has been modified illegally. The challenge is that the packets in the network are linearly combined by routers and the new packets also need to be authenticated. All current solutions for secure network coding rely on some Homomorphic property of the underlying cryptographic techniques. Uses on-the-fly verification technique which employs a classical Homomorphic hash function to address the security aspects of Storage Cloud's. These new protocols derived from our generic construction of public-key exchange protocols. Notably, we design the first publicly verifiable secure cloud storage protocol which is secure in the standard model, i.e., without modeling a hash function is a random function when arguing for the security of the protocol. Moreover, we extend our generic construction to support advanced functionalities, in particular, user anonymity, and third-party public auditing without the need of having data locally. Usage of network coding in Storage Cloud's offers benefits in terms of better bandwidth, transmission power, and delays. Prior systems employs a classical Homomorphic hash function to address the security aspects of Storage Cloud's which tends to increase latency with its hash signature size. Presents an optimization technique based on compressed hashes to reducing the data integrity verification latency between user and the cloud. This latency can be significant because of two reasons. Reducing latency for this integrity verification processing is important because of scalability and resource factors. For better performance in terms of speed and computations we suggest to use compressed hashing than classical Homomorphic hashes. By using compressed hashes, a integrity verification query process can reduce the number of bits broadcasted across index matchers, reduction of the amount of computation per lookup. The cost is the processing time for compression and decompression, which can use simple arithmetic coding, and less memory use at the query processing thread, that previously utilized the larger uncompressed form of a hashes which can be used batch verifications.

2. Literature Survey

[1]"**A Novel Cryptographic and Steganographic Approach for Secure Cloud Data Migration**".

The flexibility to store unlimited data without any worry about storage limitations available at our disposal and the freedom to use it as and when required from anywhere

in the world makes cloud computing the most preferred technology & platform to store and transfer data. Organizations and individual users are now very much comfortable to let their all-important data and software reside on the cloud servers and make themselves free from all the concerns of storage and security. However, every flexibility or benefits comes at a price and cloud computing too is not an exception. The threat of user's privacy, data confidentiality & integrity and data safety are always looming around. Among all of these, the secure transfer of data from organization's premises to the cloud servers is of utmost importance. So many encryption techniques and algorithms have been proposed by researchers in recent times to move data securely from their end to the servers. In this research paper, we propose a design for cloud architecture which ensures secure data transmission from the client's organization to the servers of the Cloud Service provider (CSP). We have used a combined approach of cryptography and steganography because it will provide a two way security to the data being transmitted on the network. First, the data gets converted into a coded format through the use of encryption algorithm and then this coded format data is again converted into a rough image through the use of steganography. Moreover, steganography also hides the existence of the message, thereby ensuring that the chances of data being tampered are minimal.

[2] "Secret Data Sharing in Cloud Environment Using Steganography and Encryption Using GA".

Cloud Computing is the next step in the evolution of on-demand network access to a shared pool of configurable resources that can be rapidly provisioned and released with minimal management effort or service provider interaction where Infrastructure, Platform and Software can be accessed as a service. The clients accessing the service, pay for what they use. Cloud Computing provides benefits in terms of low cost and accessibility of data, but its unique aspect is its security. Sharing of data is an important functionality in cloud storage. In cloud computing environment data sharing and transfer has increased exponentially. Security, integrity, non-repudiation, confidentiality, and authentication services are the most important factors in data-security. Maintaining Confidentiality and Security for critical data are highly challenging, especially when these data are stored in memory or send through the communication networks. The confidential data are embedded steganography. Data encryption technique tries to convert data to another data that is hard to understand. In this paper, a crypto-stego methodology has been proposed where image steganography and a new method of cryptographic technique is used. The steganographic technique embedded confidential data using Pixel Mapping Method (PMM), but in a chaotic sequence generated by chaotic map technique. The encryption and decryption uses Genetic Algorithm (GA) which is used to produce a cryptographic method with the help of the powerful features of the Crossover and Mutation operations of GA. Both the encryption and steganography process use secret session key which are

generated using the combination of some universal feature of cover image and the users user's secret key.

[3]"Secure Cloud Storage Meets with Secure Network Coding"

This paper reveals an intrinsic relationship between secure cloud storage and secure network coding for the first time. Secure cloud storage was proposed only recently while secure network coding has been studied for more than ten years. Although the two areas are quite different in their nature and are studied independently, we show how to construct a secure cloud storage protocol given any secure network coding protocol. This gives rise to a systematic way to construct secure cloud storage protocols. Our construction is secure under a definition which captures the real world usage of the cloud storage. Furthermore, we propose two specific secure cloud storage protocols based on two recent secure network coding protocols. In particular, we obtain the first publicly verifiable secure cloud storage protocol in the standard model. We also enhance the proposed generic construction to support user anonymity and third-party public auditing, which both have received considerable attention recently.

[4] "A Novel Privacy and Security Framework for the Cloud Network Services"

We reveal a relationship between secure cloud storage and secure network coding for the first time. Based on the relationship, we propose a systematic way to construct a generic secure cloud storage protocol based on any secure network coding protocol. As a result, we obtain the first publicly verifiable secure cloud storage protocol which is secure without using the random oracle heuristic. Further, we enhance our generic construction to support user anonymity and third-party public auditing. We hope our open sourced prototype can make a step towards practical use of secure cloud storage protocols. For future work, it is interesting to design new and efficient secure cloud storage protocols based on our generic construction and existing/ future researches on secure network coding protocols. It is also interesting to study the reverse direction, i.e., under what conditions a secure network coding protocol can be constructed from a secure cloud storage protocol. This possibly requires the latter to have some additional properties.

[5] "Securing Cryptographic Keys in the IaaS Cloud Model"

Infrastructure-as-a-Service (IaaS) is a widespread cloud computing provisioning model where ICT infrastructure, including servers, storage and networking, is supplied on-demand, in a pay-as-you-go fashion. IaaS cloud providers give their clients virtual machines (VMs) that are controlled by cloud administrators who can run, stop, restore and migrate the VMs. A typical threat to IaaS is unauthorized access of untrustworthy administrators to cloud users' sensitive information residing in VMs' memory. In this paper we focus on the threat of users' cryptographic keys being stolen from the RAM of the VM they provision. We propose a decrypt scatter/gather-decrypt technique

that allows users to carry our encryption/decryption while protecting keys from unauthorized peeks on the part of cloud administrators. Our technique does not require modification to the current cloud architecture, but only the availability of a Trusted Platform Module (TPM) capable of creating and holding a TPM protected public/private key pair. It lends itself to security-as-a-service scenarios where third parties perform encryption/decryption on behalf of data owners.

3. Related Work

Secure cloud storage problem has two main entities involved in its operations. A user and a cloud storage provider. A user outsources the data to the cloud who promises to store the data. The user then confirms the data integrity by interacting with the cloud using a secure cloud storage protocol. The motivation of data integrity checking lies in several factors. First, due to the poor management of the cloud, the user's data could be lost due to system failures (hardware or software). To cover the accident, the cloud may choose to lie to the user. Second, the cloud has a huge financial incentive to discard the data which is rarely accessed by the user. Ignoring some part of the data helps the cloud to reduce its cost. Third, a cloud could also be hacked and the data could be modified. Fourth, a cloud may behave maliciously because of various possible government pressures. Without a secure cloud storage protocol, the occurrence of these incidents may be hidden by the cloud and gone unnoticed. The main feature of a secure cloud storage protocol is that the user can check the data integrity without possessing the actual data. Traditional techniques based on hash, message authentication codes (MACs), and digital signatures however require the user to store the data locally to perform integrity checks which nullifies the purpose of cloud itself. So we need a better a system that is devoid of this problem and yet supports a secure cloud storage operations for the user. Emergence of network coding allows nodes in the network to not only forward but also process the incoming independent information flows. Network coding is a routing paradigm where a router in the network sends out encoded data packets, which are a function of received data packets, instead of the traditional store-and-forward approach. Encoding can increase the network capacity for multicast tasks. Linear coding, in which a router sends out a linear combination of received data packets, is proved to be sufficient to achieve the increased capacity. This is especially useful in cooperative networks. We propose two enhanced secure cloud storage protocols that can satisfy the needs of different applications and achieve the cooperation of different network entities. The basic idea is that each codeword in the network needs to be authenticated by checking if the codeword has been modified illegally. The challenge is that the packets in the network are linearly combined by routers and the new packets also need to be authenticated. All current solutions

for secure network coding rely on some Homomorphic property of the underlying cryptographic techniques. Uses on-the-fly verification technique which employs a classical Homomorphic hash function to address the security aspects of Storage Cloud's. These new protocols derived from our generic construction of public-key exchange protocols. Notably, we design the first publicly verifiable secure cloud storage protocol which is secure in the standard model, i.e., without modeling a hash function is a random function when arguing for the security of the protocol. Moreover, we extend our generic construction to support advanced functionalities, in particular, user anonymity, and third-party public auditing without the need of having data locally.

4. Proposed Approach

Emergence of network coding allows nodes in the network to not only forward but also process the incoming independent information flows. Network coding is a routing paradigm where a router in the network sends out encoded data packets, which are a function of received data packets, instead of the traditional store-and-forward approach. Encoding can increase the network capacity for multicast tasks. Linear coding, in which a router sends out a linear combination of received data packets, is proved to be sufficient to achieve the increased capacity. This is especially useful in cooperative networks. We propose two enhanced secure cloud storage protocols that can satisfy the needs of different applications and achieve the cooperation of different network entities. The basic idea is that each codeword in the network needs to be authenticated by checking if the codeword has been modified illegally. The challenge is that the packets in the network are linearly combined by routers and the new packets also need to be authenticated. All current solutions for secure network coding rely on some Homomorphic property of the underlying cryptographic techniques. Uses on-the-fly verification technique which employs a classical Homomorphic hash function to address the security aspects of Storage Cloud's. These new protocols derived from our generic construction of public-key exchange protocols. Notably, we design the first publicly verifiable secure cloud storage protocol which is secure in the standard model, i.e., without modeling a hash function is a random function when arguing for the security of the protocol. Moreover, we extend our generic construction to support advanced functionalities, in particular, user anonymity, and third-party public auditing without the need of having data locally. Usage of network coding in Storage Cloud's offers benefits in terms of better bandwidth, transmission power, and delays. Prior systems employs a classical Homomorphic hash function to address the security aspects of Storage Cloud's which tends to increase latency with its hash signature size. Presents an optimization technique based on compressed hashes to reducing the data integrity verification latency between user and the cloud. This latency can be significant because of two reasons. Reducing latency for this integrity

verification processing is important because of scalability and resource factors. For better performance in terms of speed and computations we suggest to use compressed hashing than classical Homomorphic hashes. By using compressed hashes, a integrity verification query process can reduce the number of bits broadcasted across index matchers, reduction of the amount of computation per lookup. The cost is the processing time for compression and decompression, which can use simple arithmetic coding, and less memory use at the query processing thread, that previously utilized the larger uncompressed form of a hashes which can be used batch verifications.

5. System Architecture

r

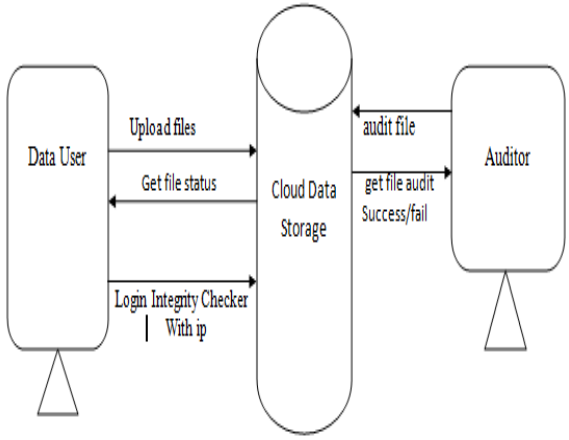

Fig: - Architecture

Uses on-the-fly verification technique which employs a classical Homomorphic hash function to address the security aspects of Storage Cloud's. These new protocols derived from our generic construction of public-key exchange protocols. Notably, we design the first publicly verifiable secure cloud storage protocol which is secure in the standard model, i.e., without modeling a hash function is a random function when arguing for the security of the protocol.

6. Compressed Hashing

Recent studies have shown that hashing methods are effective for high dimensional nearest neighbor search. A common problem shared by many existing hashing methods is that in order to achieve a satisfied performance, a large number of hash tables (i.e., long code words) are required. To address this challenge, in this paper we propose a novel approach called Compressed Hashing by exploring the techniques of Secure Network Coding and compressed sensing. In particular, we introduce a Secure Network Coding scheme, based on the approximation theory of integral operator that generates sparse representation for high dimensional vectors. We have developed a Compressed Hashing algorithm for high dimensional nearest neighbor search by combining the techniques of Secure Network Coding and compressed sensing.

The key idea is to first generate compact Secure Network codes based on the theory of density function estimation for high dimensional vectors that preserve the relationship between the data points, and then project secure network vectors into low dimensional space to preserve pair wise distances by exploring the RIP condition.

Algorithm 1 Compressed Hashing

Input:

$\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$: the database;
K: the number of bits for hashing codes;
m: the number of anchor points;
$h > 0$: the kernel width used by RBF;
s: the number of nearest anchors in sparse coding;

1: Apply k-means to compute m cluster centers from the data points in $\mathcal{D}$, and use them as the anchor points $V \in \mathbb{R}^{m \times d}$.

2: Generate sparse representation $Z \in \mathbb{R}^{n \times m}$ for data points in $\mathcal{D}$, based on the anchor points in V, using Eq.(5).

3: Generate linear projections $\Phi \in \mathbb{R}^{m \times K}$ by drawing $\Phi_{j,k}$ from $\mathcal{N}(0, 1/K)$ independently. Compute the embedding of data by $Y' = Z\Phi$.

4: Compute the hashing code Y by thresholding $Y'_{i,k}$ with respect to the median $\bar{y}_k$.

Output:

The model:
The anchor points: $\{\hat{\mathbf{x}}_i\}_{i=1}^{m}, \hat{\mathbf{x}}_i \in \mathbb{R}^d$;
The random projection matrix: $\Phi \in \mathbb{R}^{m \times K}$; Binary hashing codes for the training samples: $Y \in \{0, 1\}^{N \times K}$

Empirical studies on the large data sets show that the proposed Compressed Hashing algorithms scale well to data size and significantly outperforms the state-of-the-art hashing methods in retrieval accuracy.

7. Results

Audit Check	Processing Time (in sec)
Existing	1.16
Proposed	1.053
Enhanced	0.85

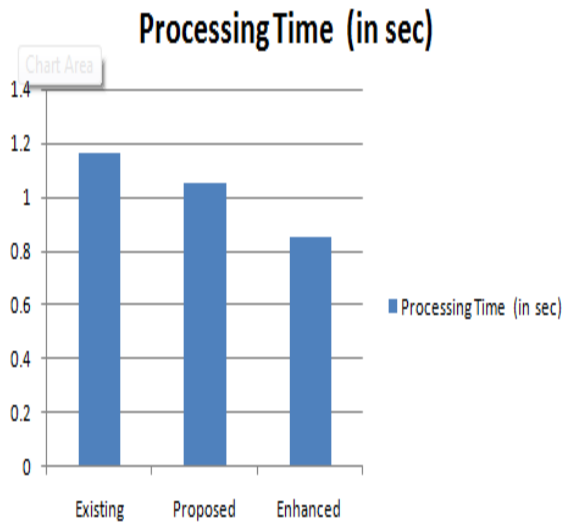

Fig:- Audit checking processing

8. Conclusion

With the knowledge of relationship between the secure cloud storage and secure network coding, this Homomorphic scheme helps in increasing the security for the files of user. We used Homomorphic mechanism to make the security stronger. It provides security in addition to detecting pollution attacks. Using Homomorphic Scheme only the authenticated user can decrypt the data. By using this technique we get the feasible time for both encryption an also the decryption process which helps the users to upload as well download the files in a stipulated time

9. References

[1] Fey Chen, Tao Xiang, Yuanyuan yang, and Sherman S.M. Chow "Secure cloud storage meets with secure network coding" in Proc. IEEE Transactions on computers, Vol.65, No 6. 2016

[2] Q. Li, J. C. Luis, and D.-M. Chiu, "On the security and efficiency of content

distribution via network coding," IEEE Trans. Dependable Secure Comput., vol. 9, no. 2, pp. 211–221, Mar./Apr. 2012.

[3] S. Agrawal and D. Bone, "Homomorphic maces: Mac-based integrity for network coding," in Proc. Int. Conf. Appl. Cryptography Newt. Security, 2009, pp. 292–305.

[4] F. Zhao, T. Talker, M. M_edard, and K. J. Han, "Signatures for content distribution with network coding," in Proc. IEEE Int. Symp. Inf. Theory, 2007, pp. 556–560.

[5] D. Charles, K. Jain, and K. Lauter,"Signatures for network coding," Int. J. Inf. Coding Theory, vol. 1, no. 1, pp. 3–14, 2009.

[6] Nuttapong Attrapadung, and Benut Liberty, "Homomorphic network coding signatures in the standard model"