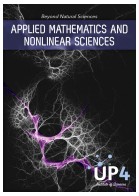
Applied Mathematics and Nonlinear Sciences 1(1) (2016) 145–158



Applied Mathematics and Nonlinear Sciences

http://journals.up4sciences.org

Overview of current techniques in remote data auditing

Hualong Wu, Bo Zhao[†]

School of Information and Technology, Yunnan, Normal University, Kunming 650500, China

Submission Info

Communicated by Wei Gao
Received 1st November 2015
Accepted 24th January 2016
Available online 28th January 2016

Abstract

The emergence of cloud computing brings the infinite imagination space, both in individual and organizations, due to its unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk. Many of the individuals or organizations ease the pressure on their local data storage, and mitigate the maintenance overhead of local data storage by using outsource data to cloud. However, the data outsourcing is not absolutely safe in the cloud. In order to enhance the users' confidence of the integrity of their outsource data in the cloud. To promote the rapid deployment of cloud data storage service and regain security assurances with outsourced data dependability, many scholars tend to design the Remote Data Auditing (RDA) technique as a new concept to enable public auditability for the outsourced data in the cloud. The RDA is a useful technique to ensure the correctness of the data outsourced to cloud servers. This paper presents a comprehensive survey on techniques of remote data auditing in cloud server. Recently, more and more remote auditing approaches are categorized into the three different classes, that is, replication-based, erasure coding-based, and network coding-based to present a taxonomy. This paper also aims to the explore major issues.

Keywords and phrases: Cloud Computing, cloud service, data auditing, network coding
2010 Mathematics Subject Classification: 68P25, 68P30

1 Introduction

Cloud Computing has been envisioned as the next-generation architecture of IT enterprise, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk, see Mell and Grance [1], Buyya et al. [2], Wang et al. [3] and [4]. With resource virtualization, cloud can deliver computing resources and services in a pay-as-you-go mode, which is envisioned to become as convenient enough to

[†]Corresponding author.
Email address: ykzb63@126.com

ISSN 2444-8656 doi:10.21042/AMNS.2016.1.00011

use as frequently as the daily-life utilities such as electricity, gas, water and telephone in the near future Mell and Grance [5], and [6, 7].

Now, many international Internet giants provide a cloud computing service to the user and cloud computing has become a tendency. But at present, cloud computing development is facing many problems among which the problem of security is the important one. The architecture of cloud data storage service is illustrated in Fig.1. Because the cloud computing service provider is a separate entity, the data stored in the cloud, in fact its equal to give up on the data of actual control (see Wang et al. [3]). As a result, many cases brought data stored in the cloud security hidden danger due to the following reasons. The first, although the infrastructures under the cloud are much more powerful and reliable than client's hardware, they still face a broad range of both internal and external threats to data integrity Subashini and Kavitha [8]. More recently, all kinds of safety accidents in Amazon, Google, and other cloud service provider intensified people's concerns, such as, Amazon S3's downtime [9], Gmail's mass email deletion incident (see Arrington and Disaster [10]), Apple Mobile Me's post-launch downtime (see Krigsman [11]). The second, driven by their own interests and other motivations for CSPs, some cloud service providers behave dishonestly toward cloud customers regarding the status of their outsourced data. Examples include cloud service providers, for monetary reasons, reputation reasons and so on Juels et al. [12]. In short, the integrity of the data stored in the cloud server is difficult to be guaranteed, even if stored in the cloud server comparing the economy. This problem may impede successful deployment of the cloud architecture, if not properly addressed.

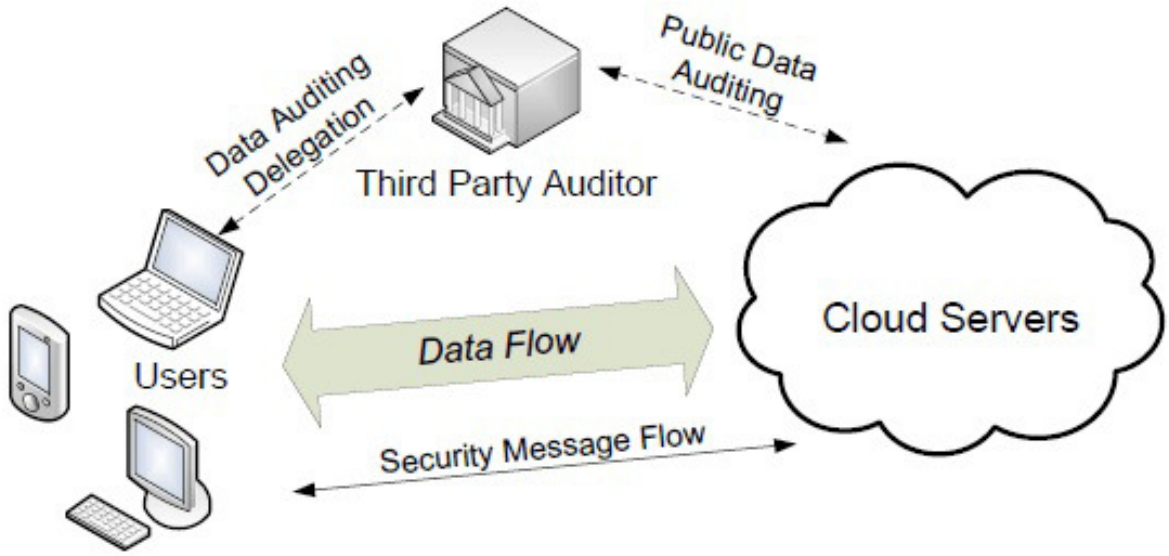


Fig. 1 The architecture of cloud data storage service.

The conventional integrity verification methods in computer cloud are inapplicable because data owners no longer physically possess the storage of their data (refer to Ateniese et al. [13]). On the other hand, downloading the whole outsource data is impractical. So, designing a suitable audit mechanism that can remotely verify the correctness of outsource data is a very necessary. To prove the correctness of the outsource data existing in cloud storage more reliably and efficiently, the remote data auditing service comprises a set of protocols designed. The remote data auditing frameworks use the technique to verify the outsourced data in which a small fragment of whole data is only required to be accessed by the auditor. The remote data auditing technique must consider

the following properties: (a) Efficiency: To audit the data with least possible computational complexity; (b) Public Verifiability: To allow delegating auditing process to a trustworthy Third Party Auditor (TPA) instead of client. (c) Frequency: To allow the verification process to be repeated as frequent as possible with different challenge messages; (d) Detection probability: It is the probability of a potential data corruption detection; (e) Recovery: The ability to restore corrupted data to original state; and (f) Dynamic Update: To still be able to audit data while the cloud user is allowed to perform delete, modify, and append operation on his/her outsourced file without requiring retrieving the entire uploaded data (refer to Wang et al. [3]).

This paper presents a basis for classifying the present and future developments within remote data auditing techniques in distributed cloud server domain and summarizes the recent remote data auditing techniques about distributions servers. This paper also analyzes the similarities and differences of the existing technology, and at the same time diagnose the significant and outstanding issues for further studies.

2 Background

The term "Cloud Computing" was inspired by the cloud symbol that is often used to show the Internet characteristic. Cloud Computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with high efficiency or service provider interaction (see [5] for more details).

The cloud service models rely on a pricing model of pay-as-you-go that charges the users on the basis of the amount of usage and some service metrics. Cloud computing, unlike traditional computing, utilizes virtualization maximize computing power. Virtualization, by separating the logical from the physical, resolves some of the challenges faced by traditional computing (see Lynch [14]). Figure 2 illustrates three models in cloud services: Software as Service (SaaS), Platform as Service (PaaS), and Infrastructure as Service (IssS).

In cloud computing, the available service models are (more details can refer to [4]):
(a) Infrastructure as a Service (IaaS). It provides the user with the capability to provision processing, storage, networks, and other fundamental computing resources, and allow the consumer to deploy and run arbitrary software, which can include operating systems and applications. The user is able to control operating operating systems, storage, deployed applications, and possibly limited control of select networking components.
(b) Platform as a Service (PaaS). It provides the consumer with the capability to deploy onto the cloud infrastructure. User created or acquired applications, produced using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but can control the deployed applications and possibly application hosting environment configurations.
(c) Software as a Service (SaaS). It provides the consumer with the capability to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices, through a thin client interface, such as a web browser (e.g. web-based e-mail). The consumer does not manage or control the underlying cloud infrastructure, including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user specific application configuration settings.
Four deployment models have been identified for cloud architecture solutions, described below:
(a) Private cloud. The cloud infrastructure is operated for a private organization. It may be managed by the organization or a third party, and may exist on premise or off premise.
(b) Community cloud. The cloud infrastructure is shared by several organizations and supports a specific community that has communal concerns (e.g., mission, security requirements, policy, and compliance considerations). It maybe managed by the organizations or a third party, and may exist on premise or off premise.
(c) Public cloud. The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.
(d) Hybrid cloud. The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities, but are bounded together by standardized or proprietary technology, which

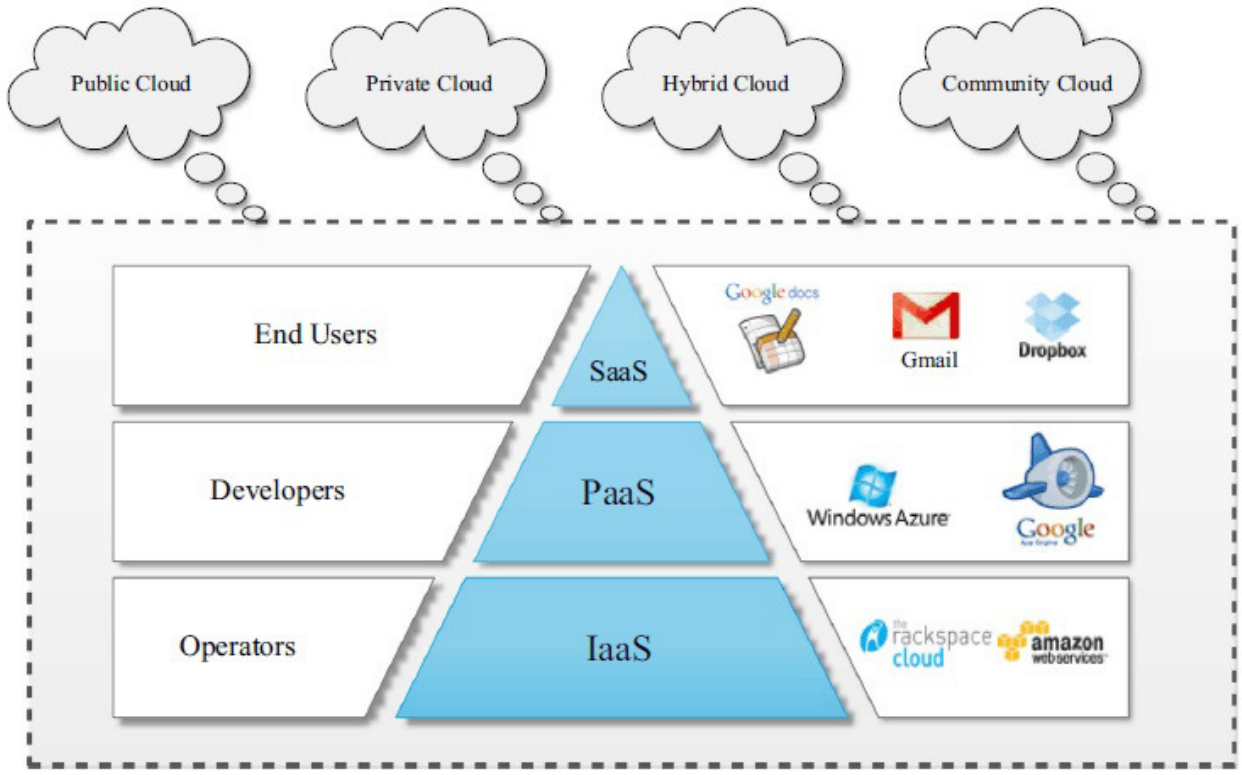


Fig. 2 Different service delivery models in cloud computing.

enables data and application portability (e.g., cloud bursting for load-balancing between clouds). Cloud computing in its advantage compared with the traditional computer, has the capability to solve a number of identified deficiencies of traditional architectures due to its unique characteristics, but the adoption of this architecture may bring a lot of problems.

3 Remote data auditing

Today, most of the individuals and organizations use cloud storage to remotely store their data and enjoy the on-demand high-quality applications and services from a shared pool of configurable computing resources, without the burden of local data storage and maintenance. Although cloud computing makes these advantages more beneficial than ever, it also brings new and challenging security threats toward users' outsourced data. Because cloud service providers (CSP) are separated administrative entities, data outsourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, this situation puts the correctness of the data in the cloud at risk due to the following reasons. (a)Although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity (see Ateniese et al. [13]). Examples of outages and security breaches of noteworthy cloud services appear from time to time (Ateniese et al. [15], Kincaid [16], and Sookhak et al., [17]). (b) There do exist various motivations for CSP to behave unfaithfully toward the cloud users because of their outsourced data status. For examples, CSP might reclaim storage for monetary reasons by discarding data that have not been or are rarely accessed, or even hide data loss incidents to maintain a reputation (Wang et al. [18], Ateniese [19], and Shah et al. [20]). In short, although outsourcing data to the cloud is economically attractive

for long-term large-scale storage, it does not immediately offer any guarantee on data integrity and availability. This problem, if not properly addressed, may impede the success of cloud architecture.

The RDA schemes for distributed cloud servers consist of four main entities. As follow: (1) Data Owner(DO): the person who uploads his/her data to the cloud space.(2) Cloud Service Provider : who has amount of computing resources and stores and manages DOs data. The CSP is also responsible for managing cloud servers. (3) Third Party Auditor (TPA): In order to alleviate the computation burden on data owners side, the auditing process is often assigned to a TPA with adequate skills and capabilities to accomplish the auditing task on behalf of the data owner. The TPA's role is particularly important when data owners possess relatively poor computing device in terms of processing power, storage space and bandwidth. While TPA is regarded as a trustful and reliable entity it might be inquisitive at the same time. Consequently, one significant countermeasure during data auditing is to prevent TPA obtaining knowledge of data owners data content and protect privacy of data. (4) User (individual or enterprise): Who is enrolled and authenticated by the DO and permitted to have pre-determined type of access on the outsourced data (refer to Sookhak et al. [17]). The architecture of RDA when TPA is involved is shown in Figure 3.

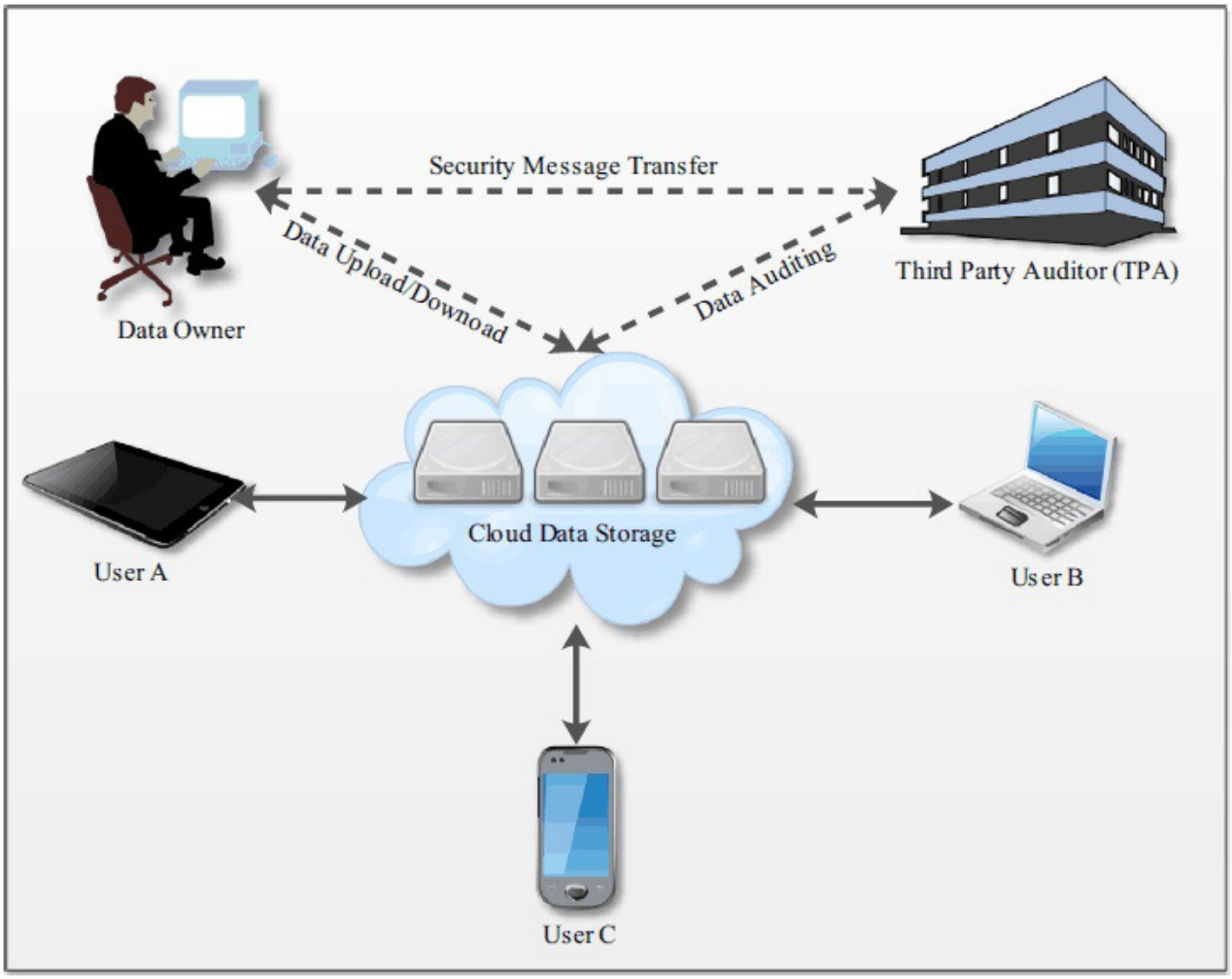


Fig. 3 Cloud data storage audit architecture.

4 The research status

In order to protect the integrity of the data, the researchers put forward many good solutions. This section introduces the current RDA methods for distributed storage systems.

4.1 Replication-based Remote Data Auditing

In an unreliable storage system, when data failures happen, redundancy plays an important role in improving the reliability. In order to effectively deal with data failures, the effective way is to use a replication technique in which multiple copies of data are outsourced within the distributed storage systems (see Ying and Vlassov [21]). When data corrupt, the client can use an intact copy of the file with size $|f|$ from any of the r unaffected servers. However, using the replication method to take up the storage space is $r|f|$ (see Agrawal and Jalote [22]).

Although realizes the method is simple, there is no strong evidence to prove that the cloud actually stores multiple copies of the data files. In other words, in the replication-based storage systems, we don't know whether our data in the server has multiple backup (see Chen et al. [23]). Such as, in peer-to-peer networks, servers perform a freeloading attack with the aim of using disproportionately more system's resources without contributing a proportionate quantity of resources back to peers (refer to Osipkov et al. [24] and Ateniese et al. [19]). As a result, the data owners encounter a decline in the durability and availability of the file and the CSP has more storage space to sell to the users.

A simple method is to allow the user to apply for a single Provable data possession (PDP) method t times for t different servers to overcome the above problems. But, the servers can collude and pretend that t copies of files are stored while only a single copy is stored in reality as an identical copy of the file is stored on all of the t servers (see Chen et al. [23]). Besides, using this method computational cost is too high and the method is inapplicable for distributed storage systems particularly for larger values of t.

Curtmola et al. proposed a provably-secure scheme, called Multiple Replica Provable Data Possession (MR-PDP), which were the first to address the collude attack in the replication-based schemes. In this scheme, the client encrypts the original file and masks the blocks of the encrypted file by using a random value (Ru) for each of the replica to generates t unique replicas (Rs) (see Barsoum and Hasan [25]). Then, the client creates a tag using a decrypted file for each block. Storage form as shown below.

$$T_i = (h(v||i).g^{b}[i])^{d} mod\ n, \tag{1}$$

(Ti is the tag for the $i^t h$ data block $(b[i])$, n is the number of blocks, d and (are the clients private key, and g is the public key)

The user selected the method of random sampling to ensure the data in the servers. By the preceding description, we may draw the method which is suitable for checking the integrity of outsourced data in distributed servers. However, the method only supports private verification and when we update the file it will generate a huge computation and communication overhead on the client and server (see Barsoum and Hasan [25]).

A scheme named Efficient Multi-Copy Provable Data Possession (EMC-PDP) was proposed by Barsoum and Hasan. This scheme is based on the technique of Boneh-Lynn-Shacham (BLS) homomorphic linear authenticators. The scheme includes two versions: Deterministic (DEMC-PDP) and Probabilistic (PEMC-PDP). The deterministic version verified the file blocks and the probabilistic version casually inspects the file.

The main method of the scheme is to generate a unique replication of file by attaching a replica number to the original file. The client using the following equation generates a tag for each block of replicas and allocation the tag among different servers.

$$T_{i,j} = (h(F_{id})u^{b[i][j]})^{d}, \tag{2}$$

(F_{id} indicates a unique fingerprint for each file that is generated by attaching the filename to the number of blocks and the number of replica, i indicates the replication number, j is the block number, d is the client's private key, and u is a generator for a bilinear group mapping G).Although this scheme support the trusted third party auditing test, to update a block of the file, the client must re-encrypt and upload the entire replicas to the

servers. Even though this scheme offering the security, its produce a higher storage overhead not only on the client but on the server.

The Dynamic Multi-Replica Provable Data Possession (DMR-PDP) scheme was proposed by Mukundan et al. [26] verified the integrity of multiple copies. The scheme utilized the technique of the Paillier encryption to generate distinct copies of the original file $F = b[1], b[2], ..., b[m]$. The process shown in figure 4 uses the following equation:

$$R_i = \{(1+N)^{b[j]}(k_i r_{i,j})\}, \tag{3}$$

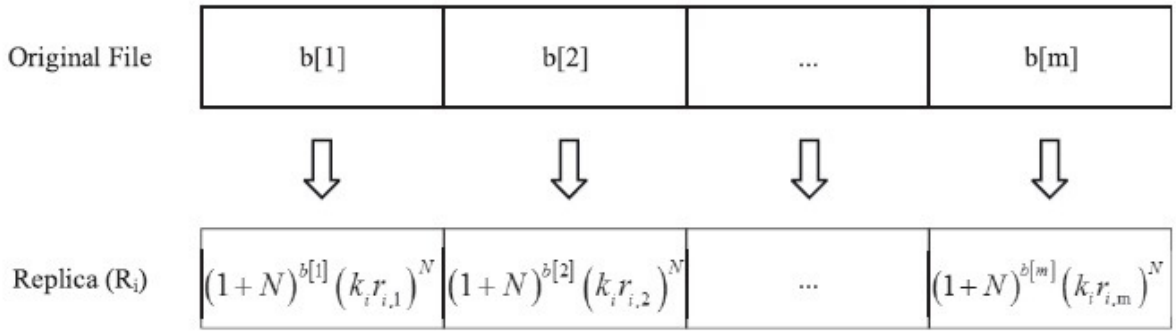


Fig. 4 Generating a Unique Replication in DMR-PDP Scheme.

(j is the index of the block, i is the number of replicas, kjrepresents a random number that is used to identify the number of replica, $r_{i,j}$ indicates a random number that is used for the Paillier encryption scheme, and N is the owner's public key)

Xiao et al. [27] using the homomorphic verification response (HVR) and hash index hierarchy (HIH), called the Cooperative Provable Data Possession (CPDP), built a replication-based remote data auditing framework for distributed systems.The HIH is a hierarchical structure includes three layers: Express, Service, and Storage Layers. They are described as follows:

1) Express Layer: offers an abstract representation of the stored resources;
2) Service Layer: offers and manages cloud storage services;
3) Storage Layer: realizes data storage on many physical devices.

Homomorphic verifiable response is used to integrate multiple responses from the different CSPs in CPCP scheme. Homomorphic verifiable response is the key technique of CPDP because it not only reduces the communication bandwidth, but also conceals the location of outsourced data in the distributed cloud storage environment. However, a heterogeneous structure of the proposed scheme leads to a high communication load due to the inter-communication between various cloud servers.

4.2 Erasure Coding-based Remote Data Auditing

The erasure code technique is based on the Maximum Distance Separable (MDS) code which is a form of data repairing technique. The erasure code technique compared to replication, provides more reliability for the same redundancy.As is shown in the Figure 5, when a block of file corruption is detected, the client can utilize the remaining intact blocks to recompute the codes of the corrupted block. Figure 5 indicates the original input file including three blocks $(b[1], b[2], b[3])$ is encrypted using the (3:2) erasure code. However, the communication overhead of this method is higher than the replication-based technique (see Kubiatowicz et al. [28], Changho and Ramchandran [29], and Li and Shu [30]).

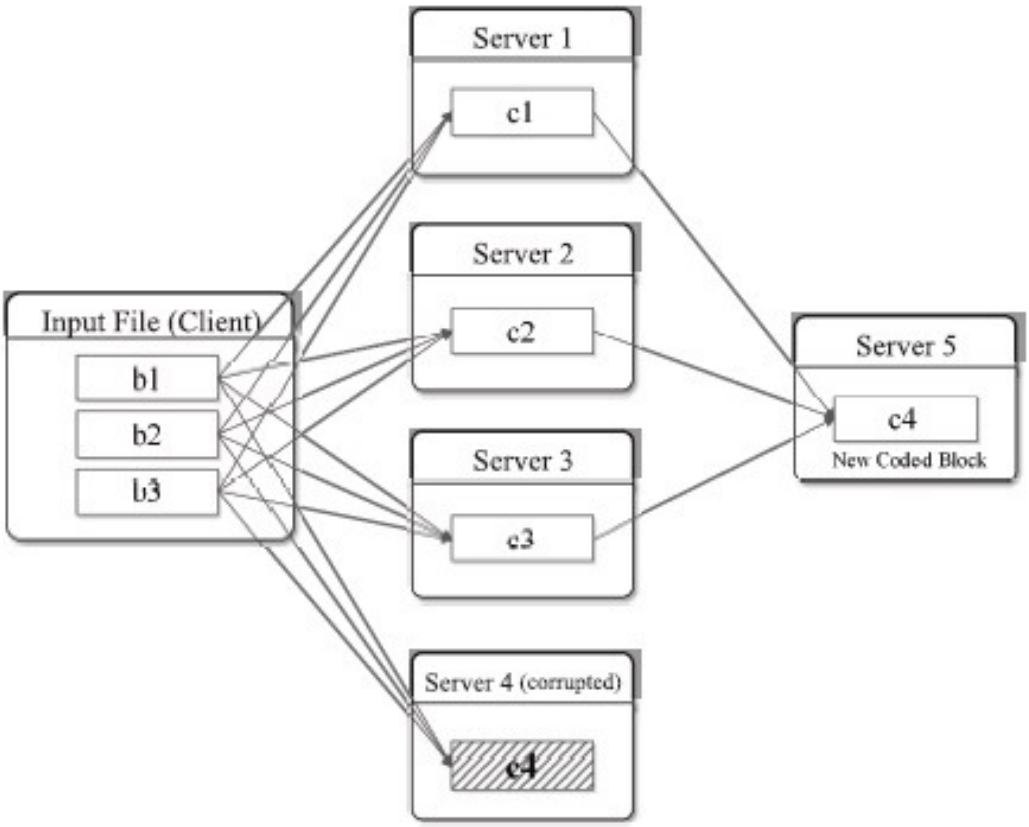


Fig. 5 Erasure Coding-Based Distributed Storage System.

The scheme of High-Availability and Integrity Layer (HAIL) is about erasure coding-based remote data auditing for distributed storage systems. When a file corruption is detected in a server, the client recovers the corrupted block using the Proofs of Retrievability scheme. This project consists of three parts as follows: (a) Dispersal code: using erasure code technique distributed the file to the servers. (b) Server code: when the servers receive the data blocks, the blocks need to be encoded using an error-correcting code to guarantee the blocks. (c) Aggregation code: the responses from all of the servers to the received challenge including the multiple Message Authentication Codes are combined into a single composite Message Authentication Code. Figure 6 shows the dispersal coding technique.

The scheme of Secure Distributed Storage (SDS) is based on Homomorphic Token and the Reed-Solomon erasure-correcting code to guarantee the data integrity and availability in an erasure code distributed storage system (see Wang [31]). Compared to previous methods, the most distinctive feature of this method is to support error localization techniques. This project mainly consists of four parts, as follows:

(a)Data Dispersal: Using the Reed-Solomon erasure code to distribute the original file to the servers and encrypt the blocks.

(b) Token Pre-computation: Before distributing the file in the servers, random subset of data blocks generate the tokens.The client randomly selects r sectors $(\{I_k\}_{k=1}^{r})$ of the j^{th} block of the extend client's file G by using pseudo-random permutation with K_{chal} as a key and generates r coefficients $\{\alpha_k\}_{K=1}^{r}$ by pseudo-random function with Kcoe as a key, to calculate the linear combination of the data block as a verification token of the block j:

$$v_j = \sum_{k=1}^{r} \alpha_i^k G^j[I_k], \tag{4}$$

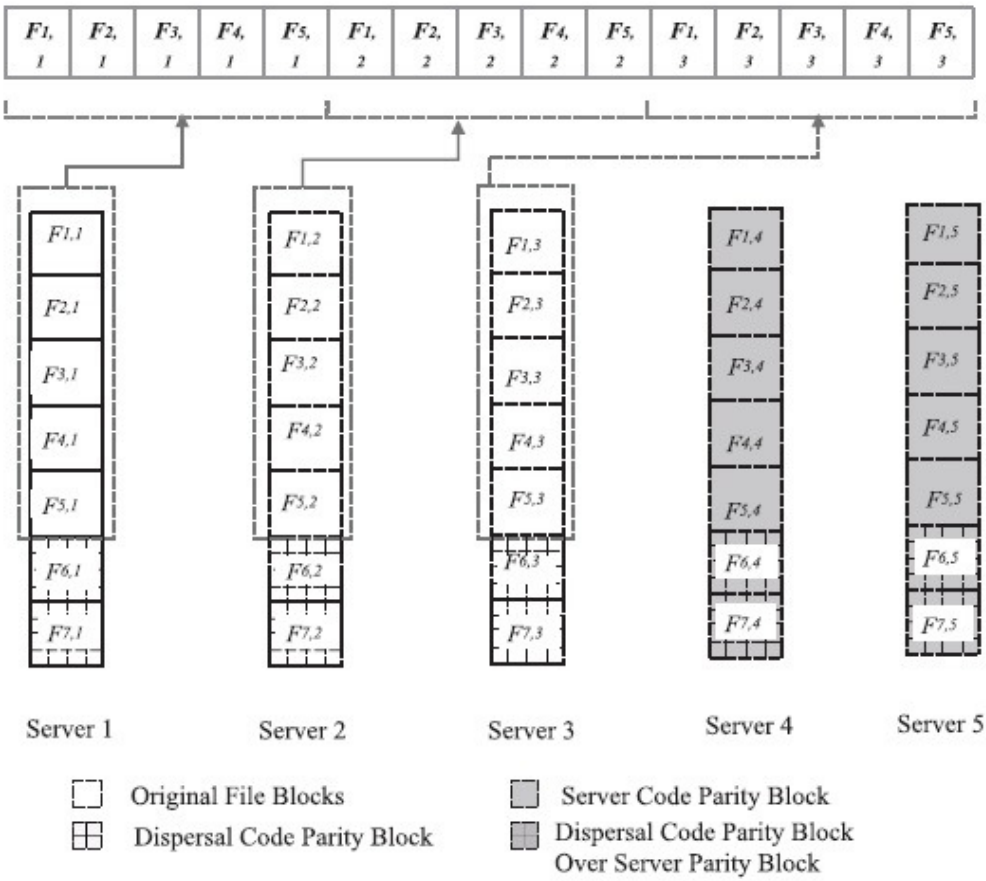


Fig. 6 Encoding of the Original File and Outsourcing to the Servers in HAIL Scheme.

(where v_j indicates the verification token of the block j and $G^j[I_k]$ is the I_k sector of j^th block of the extend client's file G)
(c) Error Localization: To identify malicious servers, the client requires to reveal the $\{\alpha_k\}_{K=1}^{r}$and the pseudo-random permutation key K_{chal} and request the servers to compute the linear combination of the certain block.
(4) Dynamic Update: The scheme supports the client dynamic update operations, such as updates, deletes, appends, and inserts. The client utilize homomorphic token updataes the changed blocks without retrieving other blocks.

4.3 Network Coding-based Remote Data Auditing

The technique of the network coding has the capability to reduce the communication overhead during the repair process. When a block failure of file was detected, a new data block is created on the basis of a linear combination of the stored data blocks across the intact servers during the repair process. For example, the client is given a file including m bloks$(F = b[1], b[2], ..., b[m])$, based on a coding coefficient vector of m random value $(v_1, v_2, ..., v_m)$:

$$NC = \sum_{i=1}^{m} v_i b[i], \tag{5}$$

the original file as a linear combination of the data blocks is generated.As is shown in Figure 7: a network coding-based distributed storage system for the original file includes three blocks $(b[1], b[2], b[3])$. The network

coding-based RDA methods must meet the following four conditions Juels et al. [12], Zhang et al. [34] and Oliveira et al. [35]:(i)Error localization (ii) Loss of a fixed layout of the file (iii) Reply attack (iv)Pollution attack.

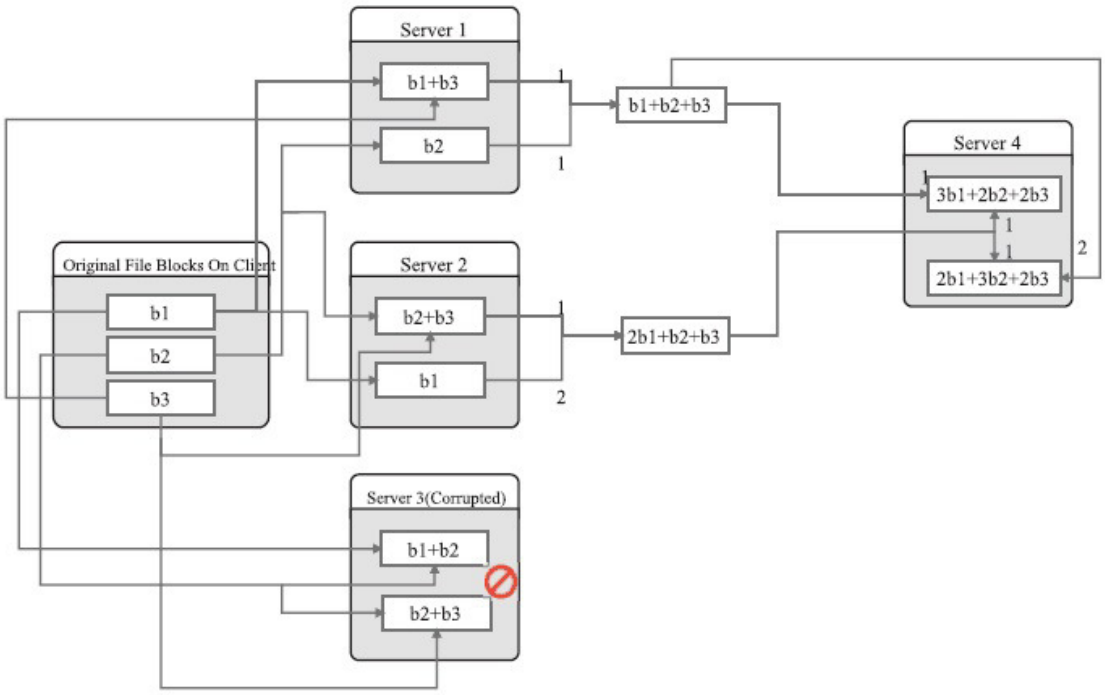


Fig. 7 Network Coding-Based Distributed Storage System.

In Anh and Markopoulou [36], the authors built on a RDA method which relies on a homomorphic Message Authentication Code scheme and a customized encryption scheme. when the client and server have a shared secret key, the scheme existence risk of pollution attacks. The homomorphic Message Authentication Code cryptosystem protect the scheme avoid the risk and this cryptosystem consists of three parts, namely: sign, combine, and verify. We use the sign algorithm to generate a tag for blocks and then use the homomorphic property to generate a linear combination of tags of the blocks. The characteristics of the scheme to some extent reduced the communication and computation overhead. In order to reduce the burden of the user on communication and computation overhead, the verification tag of the new data block is computed by combining the verification tags of intact blocks. So, we don't need to have the client generate a verification tag for the new data block. Figure 8 shows the scheme including setup, challenge, proof, and verify steps.

5 Exciting issue

The section we highlight some of the most important issues and challenges in applying and utilizing the remote data storage auditing approaches for the future research works.

5.1 Dynamic Data Update

In the static data auditing methods, the clients must download the whole outsourced data from the cloud and upload them after performing the corresponding operations, during the update operations, such as modify, delete, insert, and open. Supporting dynamic data update is an important characteristic of RDA methods both for single

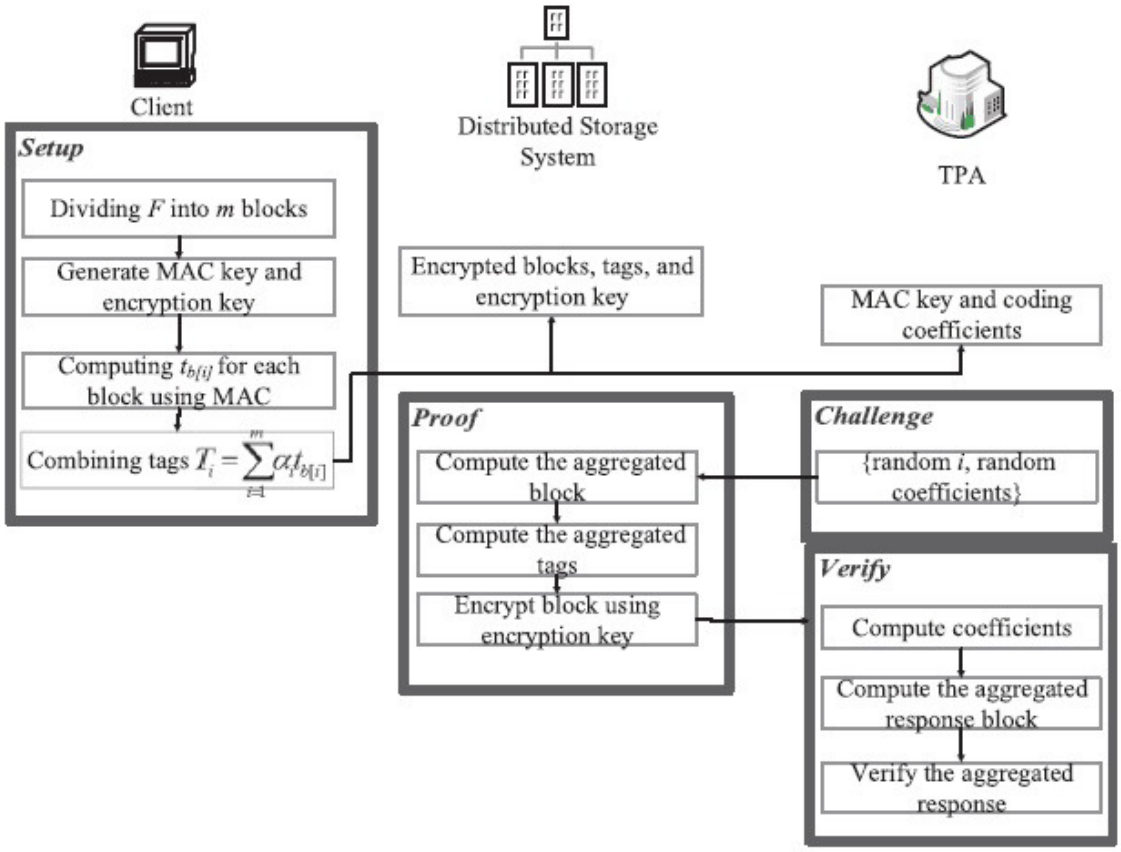


Fig. 8 Network Code Remote Data Auditing Scheme.

and distributed cloud servers, since many common applications such as online word processing intrinsically deal with dynamic form of data or involved with dynamic log files. If the auditing approach supports the only needs to download the number of blocks that are to be updated. Therefore, such a feature reduces the cost of computation and communication of updating data on the client and servers.

Although Cash et al. [37] put forward a method to overcome the dynamic data update issue in cloud computing, the lack of public verification feature makes this method impractical . On the other hand, current dynamic data update methods also impose high storage and computation overhead on data owner. As a result, enabling user to efficiently and dynamically update their outsourced data requires more future research and developments.

5.2 Batch Auditing

Because of the characteristic of the current RDA method, addressing batch auditing in the distributed storage systems is more difficult. For the TPA process multiple auditing tasks received from different users at the same time rather than performing each of the tasks one by one. In other word, batch auditing mechanism utilizes the linear sum of the random blocks to shorten the proof message and thus mitigate the corresponding communication overhead. Only a few current RDA methods focus on batch auditing issue in the distributed storage systems (refer to Agrawal and Boneh [38]). A simple way to achieve such a goal is to use a bilinear aggregate signature to combine the proof messages into a single and unique signature that can be verified by the auditor.

5.3 Privacy preserving

When data owner outsource data to the remote cloud or delegate the auditing task to the third party, under the hypothesis, the TPA is a trustworthy agent. Obviously, this is not a logical assumption increasing the possibility of leaks. Data owner does not want the TPA, of course, knowing the details of data contents (see Mandagere et al. [39] and Meyer and Bolosky [40]). How to effectively solve the problem of the development of the technology of the RDA is very important.

5.4 Large-Scale Data Auditing

In the era of big data where billions of files of data are stored in the cloud. 2013 global data volume 4.4 ZB, global data in 2014 at around 6.2 ZB, total 2015 global total data at around 8.6 ZB, 2016 will be around 12 ZB, in 2020, the amount of data will reach 40 ZB. As data volumes growing exponentially, the communication, storage and computation cost on both the auditor side and the provider side to existing RDA technology faces great challenges. Nowadays, more and more big data applications employ cloud to store a mass of data, such as Facebook. Although every day the users of Facebook just to produce a small amount of text and image data, the user behavior records are constantly updated and the user behavior is very important in the era of big data (see Sakr et al. [41] and Naone [42]). The dynamic data that a large number of data owner modifies a single bit of the outsourced data brought a problem that data auditing worsens.

6 Conclusion

In recent years, auditing outsourced data in cloud computing has gained more attention. Existing RDA approaches accomplish data checking process in diverse modes. Different models focus on different aspects, such as several approaches audit the integrity of outsourced data, a number of approaches focus on error recovery and the rest of approaches check the data ownership as well. The final goal of RDA is to guarantee the integrity and privacy of outsourced data in single and distributed cloud.

In this paper, We explained the concept of cloud computing and discussed the different techniques used to guarantee data integrity and privacy in the cloud computing. We also discussed the issues in the current RDA approaches and detailed analysis of all the methods. As we know, Cloud computing is currently developing very rapidly emerging industries and has a broad development prospects. But it's security problem that still obstruct the development of the key factors. To solve these problems, it needs the researchers to pay more attention and make more contributions.

References

[1] P. Mell and T. Grance. (2009), Draft nist working definition of cloud computing, Referenced on June.3rd, http://csrc.nist.gov/groups/SNS/cloud-computing/index.html.
[2] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, and I. Brandic. (2009), Cloud computing and emerging IT platforms: vision, hype, reality for delivering computing as the 5th utility, Future Gen. Comput. Syst., 25(6), 599-616. doi 10.1016/j.future.2008.12.001
[3] C. Wang, Q. Wang, K. Ren, and W. Lou. (2010), Privacy-preserving public auditing for data storage security in cloud computing, Proc.IEEE INFOCOM'10, Mar., 2010, San Diego, CA, Publisher: IEEE, pp: 1-9. doi 10.1109/INFCOM.2010.5462173
[4] Cloud security alliance, Top threats to cloud computing, http://www.cloudsecurityalliance.org, 2010.
[5] P. Mell and T. Grance. (2010), The NIST definition of cloud computing, Version 15, 10-7-09, http://www.wheresmyserver.co.nz/storage/media/faq-files/cloud-def-v15.pdf.
[6] Gartner: Seven cloud-computing security risks. InfoWorld. 2008-07-02. http://www.infoworld.com/d/security-central/gartner-seven-cloudcomputing-security-risks-853.
[7] Cloud security alliance. http://www.cloudsecurityalliance.org
[8] S. Subashini, V. Kavitha. (2011), A survey on security issues in service delivery models of cloud computing, Journal

of Network and Computer Applications, 34, 1-11. doi 10.1016/j.jnca.2010.07.006
[9] Amazon.com, Amazon s3 Availability Event: July 20, 2008; http:/status.aws.amazon.com/s3-20080720.html
[10] M. Arrington, G. Disaster. (2006), Reports of mass email deletions, Dec. 2006, http://www.techcrunch.com/2006/12/28/gmail-disaster-reports-of-massemail-deletions/
[11] M. Krigsman. (2008), Apple's MobileMe Experiences Post-Launch Pain, July 2008; http://blogs.zdnet.com/projectfailures/?p=908
[12] A. Juels, J. Burton, and S.Kaliski. (2007), PROs: proofs of retrievability for large files, Proc. ACM CCS 07, Oct. 2007, pp. 584-597.
[13] G. Ateniese, R. Burns, R. Curtmola, J, Herring, L. Kissner, Z. Peterson, D. Song, Provable data possession at untrusted stores, CCS'07, October 29-November 2, 2007, Alexandria, Virginia, USA. 2007, pp. 598-609.
[14] M. Lynch. (2008), The cloud wars: 100+ billion at stake, Merrill Lynch.
[15] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D. Song. (2011), Remote data checking using provable data possession, ACM Trans. Inf. Syst. Secur., 14(1)(2011)1-34, doi 10.1145/1952982.1952994
[16] J. Kincaid. (2008), MediaMax/TheLinkup closes its doors, http://www.techcrunch.com/2008/07/10/mediamaxthelinkup-closesits-doors
[17] M. Sookhak, A. Akhunzada, A. Gani, M.K. Khan, and N.B. Anuar. (2014), Towards dynamic remote data auditing in computational clouds, The Scientific World Journal, Volume 2014, Article ID 269357, 12 pages http://dx.doi.org/10.1155/2014/269357. doi 10.1155/2014/269357
[18] C. Wang, K. Wang, W. Ren, and J. Li. (2011), Enabling public auditability and data dynamics for storage security in cloud computing, IEEE Transactions on Parallel and Distributed Systems, 22(5), 847-859. doi 10.1109/TPDS.2010.183
[19] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z.Peterson, and D. Song. (2007), Provable Data Possession at Untrusted Stores, Proc. 14th ACM Conf. Computer and Comm. Security (CCS 07), 598-609, doi10.1145/1315245.1315318
[20] M.A. Shah, R. Swaminathan, and M. Baker. (2008), Privacy-preserving audit and extraction of digital contents, Cryptology ePrint Archive, Report 2008/186.
[21] Y. Liu and V. Vlassov. (2013), Replication in distributed storage systems: state of the art, possible directions, and open issues, In International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery. IEEE, 2013, pp. 225-232, doi 10.1109/CyberC.2013.44
[22] G. Agrawal and P. Jalote. (1995), Coding-based replication schemes for distributed systems, IEEE Transactions on Parallel and Distributed Systems, 6(3), 240-251, doi 10.1109/71.372774
[23] B. Chen, R. Curtmola, G. Ateniese, and R. Burns. (2010), Remote data checking for network coding-based distributed storage systems, CCSW'10, October 8, 2010, Chicago, Illinois, USA, pp. 31-42. doi 10.1145/1866835.1866842
[24] Osipkov, P. Wang, and N. Hopper. (2006), Robust accounting in decentralized P2P storage systems, In IEEE International Conference on Distributed Computing Systems, 14 pages. doi 10.1109/ICDCS.2006.71
[25] A.F. Barsoum and M.A. Hasan. (2010), Provable possession and replication of data over cloud servers, Centre For Applied Cryptographic Research (CACR), University of Waterloo, Report, 32, 1-36.
[26] R. Mukundan, S. Madria, M. Linderman, and N.Y. Rome. (2012), Replicated data integrity verification in cloud, The Bulletin of the Technical Committee on Data Engineering, pp. 55-65.
[27] D. Xiao, Y. Yang, W.B. Yao, C.H. Wu, J.Y. Liu, and Y.X. Yang. (2012), Multiple-file remote data checking for cloud storage, Computers Security, 31(2), 192-205, doi 10.1016/j.cose.2011.12.005
[28] J. Kubiatowicz, D. Bindel, Y. Chen, S. Czerwinski, P. Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao. (2000), OceanStore: an architecture for global-scale persistent storage, SIGPLAN Not., 35(11), 190-201.
[29] S. Changho and K. Ramchandran. (2010), Exact-repair MDS codes for distributed storage using interference alignment, In IEEE International Symposium on Information Theory Proceedings, pp. 161-165, doi 10.1109/ISIT.2010.5513263
[30] M.Q. Li and J.W. Shu, (2010), DACO: A high-performance disk architecture designed specially for large-scale erasure-coded storage systems, IEEE Trans. Comput., 59(10),1350-1362, doi 10.1109/TC.2010.22
[31] H. Wang. (2013), Proxy provable data possession in public clouds, IEEE Transactions on Services Computing, 6, 551-559. doi 10.1109/TSC.2012.35
[32] A.G. Dimakis, P. B. Godfrey, Y. Wu, M.J. Wainwright, and K. Ramchandran. (2010), Network coding for distributed storage systems. IEEE Transactions on Information Theory, 56(9), 4539-4551, doi 10.1109/INFCOM.2007.232
[33] A.G. Dimakis, K. Ramchandran, Y. Wu, and S. Changho. (2011), A survey on network codes for distributed storage, Proc. IEEE, 99(3), 476-489. doi 10.1109/JPROC.2010.2096170
[34] X.L. Zhang, G. Neglia, and J. Kurose. (2012), Chapter 10-Network coding in disruption tolerant networks, Academic Press, Boston, 2012, pp. 267-308, doi 10.1016/B978-0-12-380918-6.00010-X
[35] P. F. Oliveira, L. Lima, T. T. V. Vinhoza, J. Barros, and M. Medard. (2012), Coding for trusted storage in untrusted networks, IEEE Transactions on Information Forensics and Security, 7(6), 1890-1899, doi 10.1109/TIFS.2012.2217331
[36] L. Anh and A. Markopoulou. (2012), NC-Audit: Auditing for network coding storage, In International Symposium on

Network Coding (NetCod), pp. 155-160. doi 10.1109/NETCOD.2012.6261901
[37] D. Cash, A. Kupcu, and D. Wichs. (2012), Dynamic proofs of retrievability via oblivious RAM, IACR Cryptology ePrint Archive, 2012, pp. 550-550, doi 10.1007/978-3-642-38348-9_17
[38] S. Agrawal and D. Boneh. (2009), Homomorphic MACs: MAC-based integrity for network coding, In 7th International Conference on Applied Cryptography and Network Security (Lecture Notes in Computer Science), Publisher: Springer, 5536, 292-305, doi 10.1007/978-3-642-01957-9_18
[39] N. Mandagere, P. Zhou, M.A Smith, and S. Uttamchandani. (2008), Demystifying data deduplication, Demystifying Data Deduplication, 4, 1-9.
[40] D.T. Meyer and W.J. Bolosky. (2012), A study of practical deduplication, Trans. Storage, 7(4), 1-20, doi 10.1145/2078861.2078864
[41] S. Sakr, A. Liu, and A.G. Fayoumi. (2013), The family of mapreduce and large-scale data processing systems, ACM Comput. Surv., 46(1), 1-44, doi 10.1145/2522968.2522979
[42] E. Naone. (2010), What twitter learns from all those tweets, http://www. technologyreview.com/view/420968/what-twitter-learns-from-all-those-tweets/.