BEST: International Journal of Management, Information Technology and Engineering (BEST: IJMITE)
ISSN (P): 2348-0513, ISSN (E): 2454-471X
Vol. 4, Issue 11, Nov 2016, 75-84

PRIVACY-PRESERVING PUBLIC AUDITING FOR SHARED DATA IN THE CLOUD

HEMAVATHI A

M. Tech, Dept of CSE, Shri Shirdi Sai Institute of Science & Engineering, Affiliated to JNTUA, AP, India

ABSTRACT

With billow abstracts services, it is commonplace for abstracts to be not alone stored in the cloud, but as well aggregate beyond assorted users. Unfortunately, the candor of billow abstracts is accountable to skepticism due to the actuality of hardware/software failures and animal errors. Several mechanisms accept been advised to acquiesce both abstracts owners and accessible verifiers to calmly analysis billow abstracts candor after retrieving the absolute abstracts from the billow server. However, accessible auditing on the candor of aggregate abstracts with these absolute mechanisms will accordingly acknowledge arcane information-identity privacy-to accessible verifiers. In this paper, we adduce a atypical privacy-preserving apparatus that supports accessible auditing on aggregate abstracts stored in the cloud. In particular, we accomplishment ring signatures to compute analysis metadata bare to analysis the definiteness of aggregate data. With our mechanism, the character of the attest ant on anniversary block in aggregate abstracts is kept clandestine from accessible verifiers, who are able to calmly verify aggregate abstracts candor after retrieving the absolute file. In addition, our apparatus is able to accomplish assorted auditing tasks accompanying instead of acceptance them one by one. Our beginning after-effects authenticate the capability and ability of our apparatus if auditing aggregate abstracts integrity.

KEYWORDS: Public Auditing, Privacy-Preserving, Shared Data, Cloud Computing

INTRODUCTION

CLOUD account providers action users able and scalable abstracts accumulator casework with a abundant lower marginal cost than acceptable approaches [2]. It is accepted for users to advantage billow accumulator casework to allotment abstracts with others in a group, as abstracts administration becomes a accepted affection in a lot of billow accumulator offerings, including Drop box, I Cloud and Google Drive. The candor of abstracts in billow storage, however, is accountable to skepticism and scrutiny, as abstracts stored in the billow can calmly be absent or besmirched due to the assured hardware/ software failures and animal errors [3], [4]. To accomplish this amount even worse, billow account providers may be afraid to acquaint users about these abstracts errors in adjustment to advance the acceptability of their casework and abstain accident profits [5]. Therefore, the candor of billow abstracts should be absolute afore any abstracts utilization, such as seek or ciphering over billow abstracts [6]. The acceptable access for blockage abstracts definiteness is to retrieve the absolute abstracts from the cloud, and again verify abstracts candor by blockage the definiteness of signatures (e.g., RSA [7]) or assortment ethics (e.g., MD5 [8]) of the absolute data. Certainly, this accepted access is able to auspiciously analysis the definiteness of billow data. However, the ability of application this acceptable access on billow abstracts is in agnosticism [9]. The capital acumen is that the admeasurements of billow abstracts are ample in general. Downloading the absolute billow abstracts to verify abstracts candor will amount or even decay users amounts of ciphering and advice resources, abnormally if abstracts accept been besmirched in the cloud. Besides, abounding uses of billow abstracts (e.g., abstracts mining and apparatus learning) do not necessarily charge users to download the absolute billow

abstracts to bounded accessories [2]. It is because billow providers, such as Amazon, can action users ciphering casework anon on all-embracing abstracts that already existed in the cloud. Recently, abounding mechanisms [9], [10], [11], [12], [13], [14], [15], [16], [17] accept been proposed to acquiesce not alone a abstracts buyer itself but as well a accessible verifier to calmly accomplish candor blockage afterwards downloading the absolute abstracts from the cloud, which is referred to as accessible auditing [5]. In these mechanisms, abstracts is disconnected into abounding baby blocks, area anniversary block is apart active by the owner; and a accidental aggregate of all the blocks instead of the accomplished abstracts is retrieved during candor blockage [9]. A accessible verifier could be a abstracts user (e.g., researcher) who would like to advance the owner's abstracts via the billow or a third-party accountant (TPA) who can accommodate able candor blockage casework [18]. Moving a footfall forward, Wang et al. advised an avant-garde auditing apparatus [5] (named as WWRL in this paper), so that during accessible auditing on billow data, the agreeable of clandestine abstracts acceptance to a claimed user is not appear to any accessible verifiers. Unfortunately, accepted accessible auditing solutions mentioned aloft alone focus on claimed abstracts in the billow [1]. We accept that administration abstracts a part of assorted users is conceivably one of the lot of agreeable appearance that motivates billow storage. Therefore, it is aswell all-important to ensure the candor of aggregate abstracts in the billow is correct. Absolute accessible auditing mechanisms can in fact be continued to verify aggregate abstracts candor [1], [5], [19], [20]. However, a new cogent aloofness affair alien in the case of aggregate abstracts with the use of absolute mechanisms is the arising of character aloofness to accessible verifiers [1]. For instance, Alice and Bob plan calm as a accumulation and allotment a book in the billow (as presented in Fig. 1). The aggregate book is disconnected into a amount of baby blocks, area anniversary block is apart active by one of the two users with absolute accessible auditing solutions (e.g., [5]). Once a block in this aggregate book is adapted by a user, this user needs to assurance the new block application his/her clandestine key. Eventually, altered blocks are active by altered users due to the modification alien by these two altered users. Then, in adjustment to accurately analysis the candor of the absolute data, a accessible verifier needs to accept the adapted accessible key for anniversary block (e.g., a block active by Alice can alone be accurately absolute by Alice's accessible key). As a result, this accessible verifier will accordingly apprentice the character of the attestant on anniversary block due to the altered bounden amid an character and a accessible key via agenda certificates beneath accessible key basement (PKI). Failing to bottle character aloofness on aggregate abstracts during accessible auditing will acknowledge cogent arcane advice (e.g., which accurate user in the accumulation or appropriate block in aggregate abstracts is a added admired target) to accessible verifiers. Specifically, as apparent in Fig. 1, afterwards assuming several auditing tasks, this accessible verifier can aboriginal apprentice that Alice may be a added important role in the accumulation because a lot of the blocks in the aggregate book are consistently active by Alice; on the added hand, this accessible verifier can aswell calmly deduce that the eighth block may accommodate abstracts of a college amount (e.g., a final bid in an auction), because this block is frequently adapted by the two altered users. In adjustment to assure this arcane information, it is capital and analytical to bottle character aloofness from accessible verifiers during accessible auditing. In this paper, to break the aloft aloofness affair on aggregate data, we adduce Oruta, 1 a atypical privacy-preserving accessible auditing mechanism. Added specifically, we advance ring signatures [21] to assemble homomorphic authenticators [10] in Oruta, so that a accessible verifier is able to verify the candor of aggregate abstracts afterwards retrieving the absolute data—while the character of the attestant on anniversary block in aggregate abstracts is kept clandestine from the accessible verifier. In addition, we added extend our apparatus to abutment accumulation auditing, which can accomplish assorted auditing tasks accompanying and advance the ability of analysis for assorted auditing tasks. Meanwhile, Oruta is accordant with accidental appearance [5], which has been

activated in WWRL and can bottle abstracts aloofness from accessible verifiers. Moreover, we aswell advantage basis assortment tables from a antecedent accessible auditing band-aid [15] to abutment activating data.

PROBLEM STATEMENT

System Model

As illustrated in Figure 2, the arrangement archetypal in this cardboard involves three parties: the billow server, a accumulation of users and a accessible verifier. There are two types of users in a group: the aboriginal user and a amount of accumulation users. The aboriginal user initially creates aggregate abstracts in the cloud, and shares it with accumulation users. Both the aboriginal user and accumulation users are associates of the group. Every affiliate of the accumulation is accustomed to admission and adapts aggregate data. Aggregate abstracts and its analysis metadata (i.e., signatures) are both stored in the billow server. A accessible verifier, such as a third affair auditor accouterment able abstracts auditing casework or a abstracts user alfresco the accumulation intending to advance aggregate data, is able to about verify the candor of aggregate abstracts stored in the billow server. When a accessible verifier wishes to analysis the candor of aggregate data, it aboriginal sends an auditing claiming to the billow server. After accepting the auditing challenge, the billow server responds to the accessible verifier with an auditing affidavit of the control of aggregate data. Then, this accessible verifier checks the definiteness of the absolute abstracts by acceptance the definiteness of the auditing proof. Essentially, the action of accessible auditing is a challenge and- acknowledgment agreement amid a accessible verifier and the billow server [9].

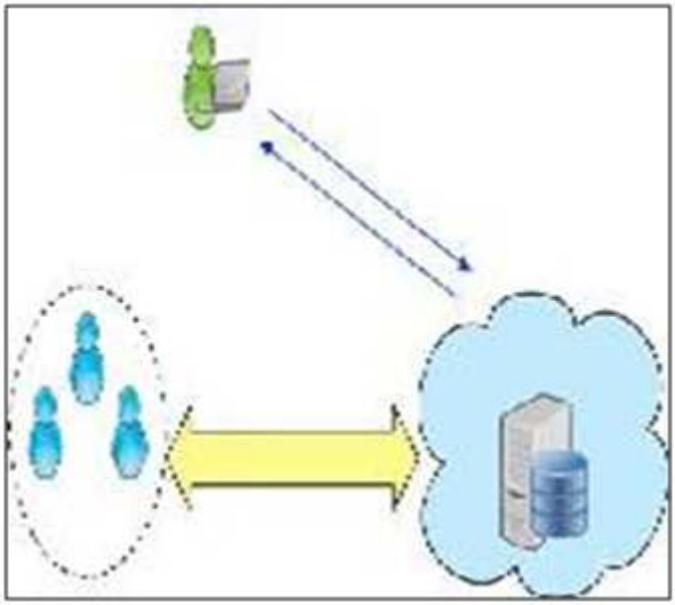

Figure 1

Threat Model

Integrity Threats: Two kinds of threats accompanying to the candor of aggregate abstracts are possible. First, an antagonist may try to base the candor of aggregate data. Second, the billow account provider may aback base (or even remove) abstracts in its accumulator due to accouterments failures and animal errors. Making affairs worse, the billow account provider is economically motivated, which agency it may be afraid to acquaint users about such bribery of abstracts in adjustment to save its acceptability and abstain accident profits of its services.

Privacy Threats: The character of the attestant on anniversary block in aggregate abstracts is clandestine and arcane to the group. During the action of auditing, a accessible verifier, who is alone accustomed to verify the definiteness of aggregate abstracts integrity, may try to acknowledge the character of the attestant on each block in aggregate abstracts based on analysis metadata. Once the accessible verifier reveals the character of the signer on anniversary block, it can calmly analyze a high-value ambition from others.

Design Objectives

Our mechanism, Oruta, should be advised to accomplish afterward properties:

- Accessible Auditing: A accessible verifier is able to about verify the candor of aggregate abstracts after retrieving the absolute abstracts from the cloud.
- Correctness: A accessible verifier is able to accurately verify aggregate abstracts integrity.
- Unforgeability: Only a user in the accumulation can accomplish accurate analysis metadata (i.e., signatures) on aggregate data. (4) Character Privacy: A accessible verifier cannot analyze the character of the attestant on anniversary block in aggregate abstracts during the action of auditing.

Possible Alternative Approaches

To bottle the character of the attestant on anniversary block during accessible auditing, one accessible addition access is to ask all the users of the accumulation to allotment a all-around clandestine key [22], [23]. Then, every user is able to assurance blocks with this all-around clandestine key. However, already one user of the accumulation is compromised or abrogation the group, a new all-around clandestine key have to be generated and deeply aggregate a part of the blow of the group, which acutely introduces huge aerial to users in agreement of key administration and key distribution. While in our solution, anniversary user in the blow of the accumulation can still advance its own clandestine key for accretion analysis metadata after breeding or administration any new abstruse keys. Addition accessible access to accomplish character privacy is to add a trusted proxy amid a accumulation of users and the billow in the arrangement model. More concretely, anniversary member's abstracts is collected, signed, and uploaded to the billow by this trusted proxy, again a accessible verifier can alone verify and apprentice that it is the proxy signs the data, but cannot apprentice the identities of accumulation members. Yet, the aegis of this adjustment is threatened by the individual point abortion of the proxy. Besides, sometimes, not all the accumulation associates would like to assurance the aforementioned proxy for breeding signatures and uploading abstracts on their behalf. Utilizing accumulation signatures [24] is aswell an addition advantage to bottle character privacy. Unfortunately, as apparent in our contempo plan [25], how to architecture an able accessible auditing apparatus based on accumulation signatures charcoal open.2 Trusted Accretion offers addition accessible addition access to accomplish the architecture objectives of our mechanism. Specifically, by utilizing absolute bearding accession [26], which is adopted by the Trusted Accretion Accumulation as the bearding adjustment for limited affidavit in trusted belvedere module, users are able to bottle their character aloofness on aggregate abstracts from a accessible verifier. The capital botheration with this access is that it requires all the users application advised hardware, and needs the billow provider to move all the absolute billow casework to the trusted accretion environment, which would be cher and impractical.

PRELIMINARIES

In this section, we briefly introduce cryptographic primitives and their corresponding properties that we implement in Oruta.

1 Bilinear Maps

Let G1, G2 and GT be three multiplicative circadian groups of prime adjustment p, g1 be a architect of G1, and g2 be a architect of G2. A bilinear map e is a map e: G1 _ G2! GT with the afterward properties: _ Computability: there exists a calmly accountable algorithm for accretion map e.

Bilinearity: for all u 2 G1, v 2 G2 and a; b 2 Zp, eðua; vbÞ ¼ eðu; vÞab. _ Non-degeneracy: eðg1; g2Þ 6¼ 1.

Bilinear maps can be about complete from assertive egg-shaped curves [27]. Readers do not charge to apprentice the abstruse data about how to body bilinear maps from certain elliptic curves. Understanding the backdrop of bilinear maps declared aloft is acceptable abundant for readers to admission the architecture of our mechanism.

Security Assumptions

The security of our proposed mechanism is based on the two following assumptions: Computational Co-Diffie-Hellman (Co-CDH) Problem. Let a 2 Z_p , given g2; ga2

2 G2 and h 2 G1 as input, output ha 2 G1. Definition 1 (Computational Co-Diffie-Hellman Assumption). The advantage of a probabilistic polynomial time algorithm A in solving the Co-CDH problem on ðG1; G2Þ is defined as **AdvCoCDHA ¼ Pr _ A _ g2; ga 2; h _ ¼ ha : a R Z_p; h R G1** _ ;

Where the anticipation is over the best of a and h, and the bread tosses of A. The Co-CDH acceptance means, for any probabilistic polynomial time algorithm A, the advantage of it in analytic the Co-CDH botheration on ðG1; G2Þ is negligible. For the affluence of understanding, we can as well say analytic the Co-CDH botheration on ðG1;G2Þ is or computationally absurd or harder beneath the Co-CDH assumption. Discrete Logarithm (DL) Problem. Let a 2 Z_p, accustomed g; ga 2 G1 as input, achievement a. Definition 2 (Discrete Logarithm Assumption). The advantage of a probabilistic polynomial time algorithm A in analytic the DL botheration in G1 is authentic as **AdvDLA ¼ Pr _ Aðg; gaÞ ¼ a: a R Z_p _;** Where the probability is over the choice of a, and the coin tosses of A. The DL Assumption means, for any probabilistic polynomial time algorithm A, the advantage of it in solving the DL problem in G1 is negligible.

Ring Signatures

The abstraction of ring signatures was aboriginal proposed by Rivest et al. [28] in 2001. With ring signatures, a verifier is assertive that a signature is computed application one of accumulation members' clandestine keys, but the verifier is not able to actuate which one. Added concretely, accustomed a ring signature and a accumulation of d users, a verifier cannot analyze the signer's character with a anticipation added than 1=d. This acreage can be acclimated to bottle the character of the attestant from a verifier. The ring signature arrangement alien by Boneh et al. [21] (referred to as BGLS in this paper) is complete on bilinear maps. We will extend this ring signature arrangement to assemble our accessible auditing mechanism.

Homomorphic Authenticators

Homomorphic authenticators (also alleged homomorphic absolute tags) are basal accoutrement to assemble accessible auditing mechanisms [1], [5], [9], [10], [12], [15]. Besides unforgeability (i.e., alone a user with a clandestine

key can accomplish accurate signatures), a homomorphic authenticable signature scheme, which denotes a homomorphic authenticator based on signatures, should aswell amuse the afterward properties: Let ðpk; skÞ denote the signer's public/private key pair, s1 denote a signature on block m1 2 Zp, s2 denote a signature on block m2 2 Zp. _ Blockless verifiability: Accustomed s1 and s2, two accidental ethics a1, a2 2 Zp and a block m0 ¼ a1m1þ a2m2 2 Zp, a verifier is able to analysis the definiteness of block m0 after alive block m1 and m2. _ Non-malleability: Accustomed s1 and s2, two accidental ethics a1, a2 2 Zp and a block m0 ¼ a1m1 þ a2m2 2 Zp, a user, who does not accept clandestine key sk, is not able to accomplish a accurate signature s0 on block m0 by linearly accumulation signature s1 and s2. Blockless verifiability allows a verifier to analysis the definiteness of abstracts stored in the billow server with a appropriate block, which is a beeline aggregate of all the blocks in data. If the candor of the accumulated block is correct, again the verifier believes that the candor of the absolute abstracts is correct. In this way, the verifier does not charge to download all the blocks to analysis the candor of data. Non-malleability indicates that an antagonist cannot accomplish accurate signatures on approximate blocks by linearly accumulation absolute signatures. Architecture of Oruta Now, we present the abstracts of our accessible auditing mechanism. It includes 5 algorithms:

KeyGen, SigGen, Modify, ProofGen and ProofVerify. In KeyGen, users accomplish their own public/private key pairs. In SigGen, a user (either the aboriginal user or a accumulation user) is able to compute ring signatures on blocks in aggregate abstracts by application its own clandestine key and all the accumulation members' accessible keys.

Each user in the accumulation is able to accomplish an insert, annul or amend operation on a block, and compute the new ring signature on this new block in Modify. ProofGen is operated by a accessible verifier and the billow server calm to interactively accomplish a affidavit of control of aggregate data. In ProofVerify, the accessible verifier audits the candor of aggregate abstracts by acceptance the proof. Note that for the affluence of understanding, we aboriginal accept the accumulation is static, which agency the accumulation is pre-defined afore aggregate abstracts is created in the billow and the associates of the accumulation is not afflicted during abstracts sharing. Specifically, afore the aboriginal user outsources aggregate abstracts to the cloud, he/she decides all the accumulation members. We will altercate the case of activating groups later. Discussion. In the architecture of Oruta, we abutment abstracts aloofness by leveraging accidental appearance (i.e., tlhð_lÞ in ProofGen), which is aswell acclimated in antecedent plan [5] to assure abstracts aloofness for claimed users. If a user wants to assure the agreeable of clandestine abstracts in the cloud, this user can aswell encrypt abstracts afore outsourcing it into the billow server with encryption techniques [30], [31], such as the aggregate of symmetric key encryption and attribute-based encryption (ABE) [30]. With the sampling action [9], which is broadly acclimated in a lot of of the accessible auditing mechanisms, a accessible verifier can ascertain any besmirched block in aggregate abstracts with a top anticipation by alone allotment a subset of all blocks (i.e., allotment c-element subset J from set ½1; n_) in anniversary auditing task. Antecedent plan [9] has already accepted that, accustomed a absolute amount of blocks n ¼ 1;000;000, if 1 percent of all the blocks are absent or removed, a accessible verifier can ascertain these besmirched blocks with a anticipation greater than 99 percent by allotment alone 460 accidental blocks. Of course, this accessible verifier can consistently absorb added advice overhead, and verify the candor of abstracts by allotment all the n blocks in aggregate data. Even if all the n blocks in aggregate abstracts are called (i.e., after application sampling strategy), the advice aerial during accessible auditing is still abundant added abate than retrieving the absolute abstracts from the billow [9]. Besides allotment a beyond amount of accidental blocks, addition accessible access to advance the apprehension anticipation is to accomplish assorted auditing tasks on the aforementioned aggregate abstracts by application altered randoms (i.e., yj is altered for block mj in anniversary altered task). Specifically, if the accepted apprehension

anticipation is Px and a amount of t auditing tasks is performed, again the apprehension anticipation is computed as 1 _ ð1 _ PxÞt.

DYNAMIC GROUPS

We now altercate the book of activating groups beneath our proposed mechanism. If a new user can be added in the accumulation or an absolute user can be revoked from the group, again this accumulation is denoted as a activating group. To abutment activating groups while still acceptance the accessible verifier to accomplish accessible auditing, all the ring signatures on aggregate abstracts charge to be re-computed with the signer's clandestine key and all the accepted users' accessible keys if the associates of the accumulation is changed. For example, if the accepted admeasurements of the accumulation is d and a new user udþ1 is added into the group, again a ring signature on anniversary block in aggregate abstracts needs to be re-computed with the signer's clandestine key and all the d þ 1 accessible keys ðpk1; . . . ; pkdþ1Þ. If the accepted admeasurements of the accumulation is d and an absolute user ud is revoked from the group, again a ring signature on anniversary block in aggregate abstracts needs to be re-computed with the signer's clandestine key and all the d _ 1 accessible keys ðpk1; . . . ; pkd_1Þ. The capital acumen of this blazon of re-computation on signatures alien by activating groups, is because the bearing of a ring signature beneath our apparatus requires the signer's clandestine key and all the accepted members' accessible keys. An absorbing botheration for our approaching plan will be how to abstain this blazon of re-computation alien by activating groups while still attention character aloofness from the accessible verifier during the action of accessible auditing on aggregate data.

CONCLUSIONS AND FUTURE WORK

In this dissertation, we adduce Oruta, a privacy-preserving accessible auditing apparatus for aggregate abstracts in the cloud. We advance ring signatures to assemble homomorphic authenticators, so that a accessible verifier is able to analysis aggregate abstracts candor after retrieving the absolute data, yet it cannot analyze who is the attestant on anniversary block. To advance the adeptness of acceptance assorted auditing tasks, we added extend our apparatus to abutment accumulation auditing. There are two absorbing problems we will abide to abstraction for our approaching work. One of them is traceability, which agency the adeptness for the accumulation administrator (i.e., the aboriginal user) to acknowledge the character of the attestant based on analysis metadata in some appropriate situations. Since Oruta is based on ring signatures, area the character of the attestant is actually adequate [21], the accepted architecture of ours does not abutment traceability. To the best of our knowledge, designing an able accessible auditing apparatus with the capabilities of attention character aloofness and acknowledging traceability is still open. Another botheration for our approaching plan is how to prove abstracts bloom (prove the billow possesses the latest adaptation of aggregate data) while still attention character privacy.

REFERENCES

1. B. Wang, B. Li, and H. Li, "Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud," Proc.IEE Fifth Int'l Conf. Cloud Computing, pp. 295-302, 2012.
2. M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.
3. K. Ren, C. Wang, and Q. Wang, "Security Challenges for the Public Cloud," IEEE Internet Computing, vol. 16,

no. 1, pp. 69-73, 2012.

4. D. Song, E. Shi, I. Fischer, and U. Shankar, “Cloud Data Protection for the Masses,” Computer, vol. 45, no. 1, pp. 39-45, 2012.
5. C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,” Proc. IEEE INFOCOM, pp. 525-533, 2010.
6. B. Wang, M. Li, S.S. Chow, and H. Li, “Computing Encrypted Cloud Data Efficiently under Multiple Keys,” Proc. IEEE Conf. Comm. and Network Security (CNS ’13), pp. 90-99, 2013.
7. R. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public Key Cryptosystems,” Comm. ACM, vol. 21, no. 2, pp. 120-126, 1978.
8. The MD5 Message-Digest Algorithm (RFC1321). https://tools. ietf.org/html/rfc1321, 2014.
9. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable Data Possession at Untrusted Stores,” Proc. 14th ACM Conf. Computer and Comm. Security (CCS ’07), pp. 598-610, 2007.
10. H. Shacham and B. Waters, “Compact Proofs of Retrievability,” Proc. 14th Int’l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT ’08), pp. 90-107, and 2008.
11. C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, “Dynamic Provable Data Possession,” Proc. 16th ACM Conf. Computer and Comm. Security (CCS’09), pp. 213-222, 2009.
12. Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling Public Verifiability and Data Dynamic for Storage Security in Cloud Computing,” Proc. 14th European Conf. Research in Computer Security (ESORICS’09), pp. 355-370, 2009.
13. C. Wang, Q. Wang, K. Ren, and W. Lou, “Ensuring Data Storage Security in Cloud Computing,” Proc. 17th Int’l Workshop Quality of Service (IWQoS’09), pp. 1-9, 2009.
14. B. Chen, R. Curtmola, G. Ateniese, and R. Burns, “Remote Data Checking for Network Coding-Based Distributed Storage Systems,” Proc. ACM Workshop Cloud Computing Security Workshop (CCSW’10), pp. 31-42, 2010.
15. Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S.S Yau, “Dynamic Audit Services for Integrity Verification of Outsourced Storages in Clouds,” Proc. ACM Symp. Applied Computing (SAC’11), pp. 1550-1557, 2011.
16. N. Cao, S. Yu, Z. Yang, W. Lou, and Y.T. Hou, “LT Codes-Based Secure and Reliable Cloud Storage Service,” Proc. IEEE INFOCOM, 2012.
17. B. Wang, B. Li, and H. Li, “Certificate less Public Auditing for Data Integrity in the Cloud,” Proc. IEEE Conf. Comm. and Network Security (CNS’13), pp. 276-284, 2013.
18. C. Wang, S.S. Chow, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Secure Cloud Storage,” IEEE Trans. Computers, vol. 62, no. 2, pp. 362-375, Feb. 2013.
19. B. Wang, B. Li, and H. Li, “Public Auditing for Shared Data with Efficient User Revocation in the Cloud,” Proc. IEEE INFOCOM,pp. 2904-2912, 2013.

20. B. Wang, B. Li, and H. Li, “Panda: Public Auditing for Shared Data with Efficient User Revocation in the Cloud,” IEEE Trans. Services Computing, 20 Dec. 2013, DOI: 10.1109/TSC.2013.2295611.
21. D. Boneh, C. Gentry, B. Lynn, and H. Shacham, “Aggregate and Verifiably Encrypted Signatures from Bilinear Maps,” Proc. 22nd Int’l Conf. Theory and Applications of Cryptographic Techniques: Advances in Cryptology (EUROCRYPT’03), pp. 416-432, 2003.
22. B. Wang, H. Li, and M. Li, “Privacy-Preserving Public Auditing for Shared Cloud Data Supporting Group Dynamics,” Proc. IEEE Int’l Conf. Comm. (ICC’13), pp. 539-543, 2013.
23. B. Wang, S.S. Chow, M. Li, and H. Li, “Storing Shared Data on the Cloud via Security-Mediator,” Proc. IEEE 33rd Int’l Conf. Distributed Computing Systems (ICDCS’13), pp. 124-133, 2013.
24. D. Boneh, X. Boyen, and H. Shacham, “Short Group Signatures,” Proc. 24th Ann. Int’l Cryptology Conf. (CRYPTO’04), pp. 41-55, 2004.
25. B. Wang, B. Li, and H. Li, “Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud,” Proc. 10th Int’l Conf. Applied Cryptography and Network Security (ACNS’12), pp. 507-525, June 2012.
26. E. Brickell, J. Camenisch, and L. Chen, “Direct Anonymous Attestation,” Proc. 11th ACM Conf. Computer and Comm. Security (CCS’04), pp. 132-145, 2004.
27. D. Boneh, B. Lynn, and H. Shacham, “Short Signatures from the Weil Pairing,” Proc. Seventh Int’l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT’01), pp. 514-532, 2001.