www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 6 Issue 4 April 2017, Page No. 20851-20858
Index Copernicus value (2015): 58.10 DOI: 10.18535/ijecs/v6i4.09

DeyPoS: Deduplicatable Dynamic Proof of Storage for Multi-User Environments

Prof.Ashok Kalal, Mr. Piyush Kumar Jha, Mr. Kunal Jondhalekar, Mr. Pavan Pandey, Ms. Himanshu Thakur

Computer Department Alard College Of Engineering & Management, Pune

Computer Department Alard College Of Engineering & Management, Pune

piyushjha81@gmail.com

Computer Department Alard College Of Engineering & Management, Pune

kunal.jondhalekar@gmail.com

Computer Department Alard College Of Engineering & Management, Pune

pandeypavan03@gmail.com

Computer Department Alard College Of Engineering & Management, Pune

Hthakur5789@gmail.com

ABSTRACT :- *Dynamic Proof of Storage (PoS) is a useful cryptographic primitive that allows a user to check the integrity and efficiently update the files in a cloud server. There has been many solutions proposed for Dynamic Proof of Storage in singleuser environment but for multi user problems is still unsolvable. A multi-user cloud storage system needs the secure client side cross user deduplication technique, which allows a user to stop the uploading process and gain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server.As we know, none of the existing dynamic PoSs can support this technique. In this paper, we elaborate the concept of deduplicatable dynamic proof of storage and propose an efficient construction called DeyPoS, to achieve dynamic Proof of Storage and secure cross-user deduplication, simultaneously.To build a novel tool called Homomorphic Authenticated Tree (HAT) to address challenges such as structure diversity and private tag generation.Hence we prove the security of our construction, and the theoretical analysis and experimental results show that our research is practically valid and applicable.*

Keywords :- *Cloud storage, dynamic proof of storage, deduplication.*

1.INTRODUCTION:-

Deduplicatable Dynamic proof of storage is a part of data outsourcing which is widely used by organisations such as Amazon, Google and Microsoft.Researchers introduced Proof of Storage to check the truthfullness of the files without downloading them from the cloud server.In this scheme a tag which is associated with block verifies the integrity of that block. When a user uploads a file then he/she becomes the uploader of the file but, if uploading same file is attempted by any other user then the system stops the upload of that file and gives the access of the file which has already been uploaded by the other user.This process is done by key value matching. It solves major problems such as private tag generation.This scheme reduces unnecessary computation and provides efficient storage for cloud server.

2. LITERATURE SURVEY

2.1 2010-Cryptographic cloud storage.

Author : S. Kamara and K. Lauter

In this article ,we consider the issues of building a secure cloud storage service at highest level of public cloud infrastructure where the customer trusts service provider.Here we state that at a high level various modules that combine recent and non-standard cryptographic primitives in order to achieve our goal. We survey the advantages of such an architecture which would provide to both customers and service provider and give an overview of recent updates in cryptography motivated specifically by cloud storage.

2.2 2016-A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data.

Author : Z. Xia, X. Wang, X. Sun, and Q. Wang

Due to the increasing popularity of cloud computing, many data owners are motivated to externalise their data to cloud servers for great ease and reduced cost in data management. However, sensitive data should be encrypted before externalise for privacy requirements, which outdate data usage like keyword based document retrieval. In this paper, we present a safe multi keyword ranked search scheme over encrypted cloud data, which supports dynamic update operations like deletion and insertion of documents. Specifically, the vector space model and the widely used TF -IDF model are combined in the index construction and query generation. We build a special tree based index structure and recommend a “Greedy Depth first Search” algorithm to provide accurate multi keyword ranked search. The secure and safe kNN algorithm is used to encrypt the index and query vectors, and ensure exact relevance score calculation between encrypted index and query vectors. In order to avoid statistical attacks, deluse terms are added to the index vector for hinding search results. Due to the utilization of special tree based index structure, the proposed scheme can achieve sublinear search time and deal with the deletion and insertion of documents flexibly. various experiments are conducted to indicate the efficiency of the proposed scheme.

2.3 2013-Security and privacy in cloud computing.

Author : Z. Xiao and Y. Xiao

Recent updates have given increase in popularity and success of cloud computing. However, when externalise the data and business application to a third party causes the security and privacy issues to become a critical threat. Throughout the study at hand, the authors obtain a collective goal to provide a brief review of the current security and privacy issues in cloud environments. We have recognized five most representative security and privacy attributes (i.e., confidentiality, integrity, availability, accountability, and privacy preservability). Starting with these attributes, we present the relationships among them, the weakness that may be exploited by attackers, the threat models, as well as existing defense strategies in a cloud environment. Further research directions are previously determined for each attribute.

2.4 2015- From Security to Assurance in the Cloud: A Survey.

Author : C. A. Ardagna, R. Asal, E. Damiani, and Q. H. Vu.

The cloud computing paradigm has become a important solution for the distribution of business processes and applications. In the public cloud vision, framework, stage, and software services are arranged to tenants (i.e., customers and service providers) on a pay as you go basis. Cloud tenants can use cloud resources at low prices, and high performance and flexibility, than classical on premises resources, without having to responsible about infrastructure management. Still, cloud tenants remain anxious with the cloud’s level of service and the fancy properties their applications can count on.Recently, the research association has been concentrating on the fancy aspects of the cloud paradigm, among which cloud security stands out. Several access to security have been called and summarized in general surveys on cloud security techniques. The survey in this article concentrates on the admix between cloud security and cloud security assurance. First, we provide an analysis of the state of the art on cloud security. Then, we introduce the approach of cloud security assurance and analyze its growing brunt on cloud security approaches. Finally, we present some propositions for the development of next generation cloud security and assurance solutions.

2.5 2008-Scalable and Efficient Provable Data Possession.

Author : Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik

Storage outsourcing is a increasing popular cocept which prompts a number of security issues,most of which have been widely investigated in the past. However, Provable Data Possession (PDP) is a topic that has only recently came in the research literature. The important issue is how to efficiently and securely verify that a storage server is trustfully storing its client's (potentially very large) outsourced data. The storage server cant be trusted in terms of both security and reliability. (In other words, it might maliciously or unintentionally delete hosted data; it might also consgn it to slow or off-line storage.) The problem is provoked by the client being a small computing device with limited resources. Prior work has addressed this issue using either public key cryptography or requiring the client to externise its data in encrypted form.

We construct a highly able and provably secure PDP technique based completely on symmetric key cryptography, while not requiring any large encryption. Also, in contrast with its previous models, our PDP technique allows externalisation of dynamic data, i.e it actively supports operations, such as block modification, deletion and append.

2.6 2009-Dynamic provable data possession

Author : C. Erway, A. K ¨upc ¨u, C. Papamanthou, and R. Tamassia

We consider the problem of conveniently proving the honesty of data stored at distrusted servers. In the provable data possession (PDP) model, the client reprocesses the data and then sends it to an distrusted server for storage, while keeping a small amount of meta data. The client later asks the server to prove that the stored data has not been meddled with or removed (without downloading the exact data). However, the original PDP scheme administer only to static (or append-only) files.

We present a definitional grounwork and productive constructions for dynamic provable data possession (DPDP), which extends the PDP model to support provable refurbish to stored data. We use a new version of attested dictionaries based on rank information. The price of changing renovations is a performance change from O(1) to O(logn) (or O(nεlog n), for a file consisting of n blocks, while maintaining the same (or better, respectively) probability of immorality detection. Our experiments show that this decrease in speed is very less in practice (e.g. 415KB proof size and 30ms computational reprentation for a 1GB file). We also show how to apply our DPDP scheme to externalised file systems and version control systems (e.g. CVS).

2.7 2009-Enabling public verifiability and data dynamics for storage security in cloud computing.

Author : Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou

Cloud Computing has been anticipate as the next-generation architecture of IT Enterprise. It moves the application software and databases to the unified large data centers, where the management of the data and services cannot be fully trusted. This exclusive archetype brings about many new security challenges, which have not been cleared. This work studies the issues of assuring the honesty of data storage in Cloud Computing. In particular, we can allow a third party auditor (TPA), on favour of the cloud client, to check the integrity of the changing data saved in the cloud. The introduction of TPA removes the participation of client through the examining of whether his data stored in the cloud is indeed static, which can be important in conquering economies of scale for Cloud Computing. The support for data dynamics via the most common forms of data operation, such as block modification, insertion and deletion, is also a important step toward practicality, since services in Cloud Computing are unlimited to backup data only. While prior works on assuring distant data integrity often lacks the support of either public changing data operations, this paper achieves both. We first recognise the problems and potential security problems of direct expansions with fully changing data updates from recent works and then show how to build an elegant checking scheme for logical integration of these two arresting features in our protocol design. In particular, to achieve efficient data dynamics, we improve the Proof of Retrievability model by making changes in the classic Merkle Hash Tree (MHT) construction for block tag verification. Expanding security and performance analysis show that the proposed scheme is highly capable and provably secure.

2.8 2014- Outsourced proofs of retrievability.

Author : F. Armknecht, J.-M. Bohli, G. O. Karame, Z. Liu, and C. A. Reuter

Proofs of Retrievability (POR) are cryptographic proofs that is capable of providing a cloud provider to prove that a user can incurr his file in its entirety. POR need to be executed by the user frequently to ensure that their files saved on the cloud can be entirely accessable at any point in time. To conduct and check POR, users need to be loaded with devices that have network access, and in which computational overhead is tolerable incurred by the verification process. This hinders the large scale adoption of POR by cloud users, since most users increasingly depend on portable devices that have less computational capacity, or may not always have network access.

In this paper, we define the notion of externalise proofs of retrievability (OPOR), in which users can task an explicit auditor to perform and check POR with the service provider. We argue that the OPOR setting is to security risks that existing POR security models has not been covered.We propose a formal structure and a security model for OPOR. We then propose an epitome of OPOR which builds upon the provably secure private POR scheme due effected to Shacham and Waters and we display its security in our recommended security model. We execute a prototype depend on our solution, and assess its performance in a realistic cloud setting. Our evaluation results show that our recommendation minimizes user effort, incurrs avoidable overhead on the auditor and greatly improves over current publicly verifiable POR.

2.9 2012- A dynamic proof of retrievability (PoR) scheme with o(logn) complexity.

Author : Z. Mo, Y. Zhou, and S. Chen

Cloud storage has been acquiring popularity because its flexibilty and pay as you go manner. However, this latest type of storage model also comes with security challenges. This paper elaborates the issue of ensuring data integrity in cloud storage. In the Proof of Retrievability (PoR) model, after externalise the preprocessed data to the server, the client will remove its local copies and only store a less amount of meta data. After that, the client will ask the server to provide a evidence that its data can be downloaded correctly. However, many recent PoR works apply only to static data. The current changing version of PoR scheme has an efficiency issue. In this paper, we elaborate the static PoR scheme to changing scenario. Which means, the client can perform update operations e.g. insertion, deletion and modification. After every update, the client can still detect the data losses even if the server tries to hide them. We create a new version of trusted data structure depend on a B+ tree and a merkle hash tree. We can call it Cloud Merkle B+ tre. By merging the CMBT with the BLS signature, we recommend a dynamic version of PoR scheme. Equating with the current dynamic PoR scheme, our worst case communication complexity is O(logn) rather than O(n).

2.10 2013- Practical dynamic proofs of retrievability.

Author : E. Shi, E. Stefanov, and C. Papamanthou

Proofs of Retrievability (PoR) authorize a client to save n number blocks with a server so that later the server can estimate control of all the data in a very efficient manner. Although most of the efficient PoR schemes for static data have been build, only two changing PoR schemes present. The scheme by Stefanov et. al. (ACSAC 2012) uses a huge of amount of client storage and has a huge survey cost. The scheme by Cash is majority of hypothetical interest, as it works on Oblivious RAM (ORAM) as a black box, headed to increased practical overhead.

We recommend a changing PoR scheme with contiuous client storage whose bandwidth cost is relatively equal to a Merkle hash tree, thus being very practical. Our construction overshadows the constructions of Stefanov et. al. and Cash et. al., both hypothetically and in practice. Specifically, for n number of externalised blocks of beta bits each, editing a block requires beta+O bandwidth and O(betalog n) server computation. Surveys are also very efficient, requiring beta+O(lambda^2log n) bandwidth. We also show how to build our scheme publicly errorfree, providing the first changing PoR scheme with such a property. We finally provide a very efficient execution of our scheme.

3. EXISTING SYSTEM

Secure deduplication is a method for disposing of copy duplicates of capacity information, and gives security to them. To decrease storage room and transfer data transfer capacity in distributed storage deduplication has been a surely under- stood procedure. For that reason concurrent encryption has been widely receive for secure deduplication, basic issue of making merged encryption down to earth is to proficiently and dependably deal with a colossal number of joined keys. The fundamental thought in this paper is that we can take out copy duplicates of capacity information and farthest point the harm of stolen information on the off chance that we diminish the estimation of that stolen data to the aggressor. This paper makes the first endeavor to formally address the issue of accomplishing productive and solid key administration in secure deduplication. We first present a pattern approach in which every client holds a free ace key for scrambling the joined keys and outsourcing them. Be that as it may, such a gauge key administration plan produces a huge number of keys with the expanding number of clients and obliges clients to dedicatedly secure the expert keys. To this end, we propose Dekey, User Behavior Proling and Decoys innovation. Dekey new development in which clients don't have to deal with any keys all alone however rather safely disseminate the merged key shares over different servers for insider aggressor. As a proof of idea, we execute Dekey utilizing the Ramp mystery sharing plan and show that Dekey acquires restricted overhead in reasonable situations. Client profiling and imitations, then, fill two needs. Initial one is accepting whether information access is approved when strange data access is identified, and second one is that mistaking the aggressor for counterfeit data. We place that the blend of these security components will give remarkable levels of security to the deduplication in insider and pariah assailant.

4. PROPOSED SYSTEM

Cloud computing gives boundless virtualized plan of action to client as administra- tions over the entire web while concealing the stage and executing subtle elements. Distributed storage administration is the administration of evergreen expanding mass of information. To make information administration adaptable in distributed com- puting, deduplication has been a customary method. Information pressure strategy is utilized for dispensing with the copy duplicates of rehashed information in dis- tributed storage to decrease the information duplication. This method is utilized to speedup stockpiling use furthermore be connected to network information exchanges to lessen the quantity of bytes that must be sent. Keeping numerous information du- plicates with the comparable substance, deduplication wipes out excess information by keeping one and only physical duplicate and allude other repetitive information to that duplicate. Information deduplication happens record level and additionally square level. The copy duplicates of indistinguishable document dispose of by record level deduplication .For the square level duplication which wipes out copies pieces of information that happen in non-indistinguishable documents. In spite of the fact that information deduplication takes a considerable measure of advantages, security and also protection concerns emerge as clients' touchy information are skilled to both insider and outcast assaults. In the conventional encryption giving information privacy, is conflicting with information deduplication. Conventional encryption re- quires diverse clients to encode their information with own keys. For making the attainable deduplication and keep up the information secrecy utilized united encryption system. It encodes decodes an information duplicate with a merged key, the information's substance duplicate got by registering the cryptographic hash estimation of. After the information encryption and key era process clients hold the keys.

5. Conclusion

We proposed the first realistic deduplicatable dynamic PoS scheme which makes use of complete necessities in multiconsumer cloud storage systems and proved its security within the random oracle model. The theoretical and experimental results show that the procedure is efficient, peculiarly when the file dimension and the number of the challenged blocks are large.

6. References:

[1] S. Kamara and K. Lauter, “Cryptographic cloud storage,” in Proc. of FC, pp. 136–149, 2010.

[2] Z. Xia, X. Wang, X. Sun, and Q. Wang, “A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data,” IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 340–352, 2016.

[3] Z. Xiao and Y. Xiao, “Security and privacy in cloud computing,” IEEE Communications Surveys Tutorials, vol. 15, no. 2, pp. 843–859, 2013.

[4] C. A. Ardagna, R. Asal, E. Damiani, and Q. H. Vu, “From Security to Assurance in the Cloud: A Survey,” ACM Comput. Surv., vol. 48, no. 1, pp. 2:1–2:50, 2015.

[5] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable data possession at untrusted stores,” in Proc. of CCS, pp. 598–609, 2007.

[6] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, “Scalable and Efficient Provable Data Possession,” in Proc. of SecureComm, pp. 1–10, 2008.

[7] G. Ateniese, S. Kamara, and J. Katz, “Proofs of storage from homomorphic identification protocols,” in Proc. of ASIACRYPT, pp. 319–333, 2009.

[8] C. Erway, A. Ku¨pcu¨, C. Papamanthou, and R. Tamassia, “Dynamic provable data possession,” in Proc. of CCS, pp. 213–222, 2009.

[9] R. Tamassia, “Authenticated Data Structures,” in Proc. of ESA, pp. 2–5, 2003.

[10] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling public verifiability and data dynamics for storage security in cloud computing,” in Proc. of ESORICS, pp. 355–370, 2009.

[11] F. Armknecht, J.-M. Bohli, G. O. Karame, Z. Liu, and C. A. Reuter, “Outsourced proofs of retrievability,” in Proc. of CCS, pp. 831–843, 2014.

[12] H. Shacham and B. Waters, “Compact Proofs of Retrievability,” Journal of Cryptology, vol. 26, no. 3, pp. 442–483, 2013.

[13] Z. Mo, Y. Zhou, and S. Chen, “A dynamic proof of retrievability (PoR) scheme with o(logn) complexity,” in Proc. of ICC, pp. 912– 916, 2012.

[14] E. Shi, E. Stefanov, and C. Papamanthou, “Practical dynamic proofs of retrievability,” in Proc. of CCS, pp. 325–336, 2013.

[15] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, “Proofs of ownership in remote storage systems,” in Proc. of CCS, pp. 491– 500, 2011.

[16] J. Douceur, A. Adya, W. Bolosky, P. Simon, and M. Theimer, “Reclaiming space from duplicate files in a serverless distributed file system,” in Proc. of ICDCS, pp. 617–624, 2002.

[17] A. Juels and B. S. Kaliski, Jr., “PORs: Proofs of retrievability for large files,” in Proc. of CCS, pp. 584–597, 2007.

[18] H. Shacham and B. Waters, “Compact proofs of retrievability,” in Proc. of ASIACRYPT, pp. 90–107, 2008.

[19] Y. Dodis, S. Vadhan, and D. Wichs, “Proofs of retrievability via hardness amplification,” in Proc. of TCC, pp. 109–127, 2009.

[20] K. D. Bowers, A. Juels, and A. Oprea, “HAIL: A high-availability and integrity layer for cloud storage,” in Proc. of CCS, pp. 187–198, 2009.

[21] C. Wang,Q. Wang, K. Ren,and W.Lou, “Privacy-preserving public auditing for data storage security in cloud computing,” in Proc. of INFOCOM, pp. 1–9, 2010.

[22] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D. Song, “Remote data checking using provable data possession,” ACM Transactions on Information System Security, vol. 14, no. 1, pp. 1–34, 2011.

[23] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, “Cooperative provable data possession for integrity verification in multicloud storage,” IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231–2244, 2012.

[24] J. Xu and E.-C. Chang, “Towards efficient proofs of retrievability,” in Proc. of ASIACCS, pp. 79–80, 2012.

[25] J. Chen, L. Zhang, K. He, R. Du, and L. Wang, “Message-locked proof of ownership and retrievability with remote reparing in cloud,” Security and Communication Networks, 2016.

[26] E. Stefanov, M. van Dijk, A. Juels, and A. Oprea, "Iris: A scalable cloud file system with efficient integrity checks," in Proc. of ACSAC, pp. 229–238, 2012.

[27] D. Cash, A. Ku¨pc¸u¨, and D. Wichs, "Dynamic proofs of retrievability via oblivious RAM," in Proc. of EUROCRYPT, pp. 279–295, 2013.

[28] M. Azraoui, K. Elkhiyaoui, R.Molva, and M. ¨Onen, "StealthGuard: Proofs of Retrievability with Hidden Watchdogs," in Proc. of ESORICS, pp. 239–256, 2014.

[29] Z. Ren, L. Wang, Q. Wang, and M. Xu, "Dynamic Proofs of Retrievability for Coded Cloud Storage Systems," IEEE Transactions on Services Computing, vol. PP, no. 99, pp. 1–1, 2015.

[30] R. Di Pietro and A. Sorniotti, "Boosting Efficiency and Security in Proof of Ownership for Deduplication," in Proc. of ASIACCS, pp. 81–90, 2012.

[31] J. Xu, E.-C. Chang, and J. Zhou, "Weak leakage-resilient clientside deduplication of encrypted data in cloud storage," in Proc. of ASIACCS, pp. 195–206, 2013.

[32] S. Keelveedhi, M. Bellare, and T. Ristenpart, "DupLESS: Serveraided encryption for deduplicated storage," in Proc. of USENIX Security, pp. 179–194, 2013.

[33] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure Deduplication with Efficient and Reliable Convergent Key Management," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 6, pp. 1615–1625, 2014.

[34] J. Li, Y. K. Li, X. Chen, P. Lee, and W. Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication," IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 5, pp. 1206–1216, 2015.

[35] Q. Zheng and S. Xu, "Secure and efficient proof of storage with deduplication," in Proc. of CODASPY, pp. 1–12, 2012.

[36] R. Du, L. Deng, J. Chen, K. He, and M. Zheng, "Proofs of ownership and retrievability in cloud storage," in Proc. of TrustCom, pp. 328–335, 2014.

[37] B. Wang, B. Li, and H. Li, "Public auditing for shared data with efficient user revocation in the cloud," in Proc. of INFOCOM, pp. 2904–2912, 2013.

[38] B. Wang, B. Li, and H. Li, "Oruta: privacy-preserving public auditing for shared data in the cloud," IEEE Transactions on Cloud Computing, vol. 2, no. 1, pp. 43–56, 2014.

[39] J. Yuan and S. Yu, "Efficient public integrity checking for cloud data sharing with multi-user modification," in Proc. of INFOCOM, pp. 2121–2129, 2014.

[40] R. Gennaro and D. Wichs, "Fully Homomorphic Message Authenticators," in Proc. of ASIACRYPT, pp. 301–320, 2013.

[41] D. Boneh and D. M. Freeman, "Homomorphic Signatures for Polynomial Functions," in Proc. of EUROCRYPT, pp. 149–168, 2011.

[42] D. Catalano, D. Fiore, and B. Warinschi, "Homomorphic Signatures with Efficient Verification for Polynomial Functions," in Proc. of CRYPTO, pp. 371–389, 2014.

[43] A. Yun, J. H. Cheon, and Y. Kim, "On Homomorphic Signatures for Network Coding," IEEE Transactions on Computers, vol. 59, no. 9, pp. 1295–1296, 2010.

[44] C. Cheng and T. Jiang, "An Efficient Homomorphic MAC with Small Key Size for Authentication in Network Coding," IEEE Transactions on Computers, vol. 62, no. 10, pp. 2096–2100, 2013.

[45] J. Chen, K. He, R. Du, M. Zheng, Y. Xiang, and Q. Yuan, "Dominating Set and Network Coding-Based Routing in Wireless Mesh Networks," IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 2, pp. 423–433, 2015.

[46] D. Catalano, "Homomorphic Signatures and Message Authentication Codes," in Proc. of SCN, pp. 514–519, 2014.

[47] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-locked encryption and secure deduplication," in Proc. of EUROCRYPT, pp. 296–312, 2013.

[48] M. Abadi, D. Boneh, I. Mironov, A. Raghunathan, and G. Segev, "Message-locked encryption for lock-dependent messages," in Proc. of CRYPTO, pp. 374–391, 2013.