Security Saving open Reviewing for Recovery Code based Distributed Storage

Madhuri Parasar
VTU, Dept. CSE AMCEC
Bangalore, India

Geetha. N
VTU, Dept. CSE AMCEC
Bangalore, India

***Abstract*—To secure outsourced information in distributed storage defilements, adding adaptation to internal failure to distributed storage together with information respectability checking and disappointment reparation gets to be basic. As of late, recovering codes have picked up ubiquity due to their lower repair transmission capacity while giving adaptation to internal failure. Existing remote checking strategies for recovering coded information just give private inspecting, requiring information proprietors to dependably stay online and handle inspecting, and in addition repairing, which is at times unreasonable. In this paper, we propose an open inspecting plan for the recovering code-based distributed storage. To take care of the recovery issue of fizzled authenticators in the nonattendance of information proprietors, we present an intermediary, which is special to recover the authenticators, into the customary open reviewing framework model. In addition, we plan a novel open obvious authenticator, which is produced by two or three keys what's more, can be recovered utilizing incomplete keys. In this manner, our plan can totally discharge information proprietors from online weight. Moreover, we randomize the encode coefficients with a pseudorandom capacity to protect information security.**

Keywords - Terms—Cloud storage, regenerating codes, public audit, privacy preserving, authenticator regeneration, proxy, privileged,provable secure.

I. INTRODUCTION

Cloud storage is now gaining popularity because itoffers a flexible on-demand data outsourcing servicewith appealing benefitsDistributed storage is presently picking up fame since it offers an adaptable on-interest information outsourcing administration with engaging advantages: alleviation of the weight for capacity administration, all inclusive information access with area autonomy.

We plan a novel homomorphic authenticator taking into account BLS signature [17], which can be created by a couple of mystery keys and confirmed openly. Using the direct subspace of the recovering codes, the authenticators can be figured effectively.

- To the best of our insight, our plan is the first to permit protection saving open reviewing for regeneratingcode- based distributed storage. The coefficients are conceal by a PRF(Pseudorandom Function) amid the Setup stage to evade spillage of the first information. This technique is lightweight and does not present any computational overhead to the cloud servers or TPA.
- Our plan totally discharges information proprietors from online .load for the recovery of squares and authenticators at broken servers and it gives the benefit to an intermediary for the reparation
- Optimization measures are taken to enhance the adaptability what's more, productivity of our evaluating plan; in this manner, the capacity overhead of servers, the computational overhead of the information proprietor and correspondence overhead amid the review stage can be adequately lessened.
- Our plan is provable secure under arbitrary prophet model against foes.

In 2014---IEEE Transactions—Provable data possession at untrusted stores:-In recent years, cloud computing has gradually become the mainstream of Internet services. When cloud computing environments become more perfect, the business and user will be an enormous amount of data stored in the remote cloud storage devices, hoping to achieve random access, datacollection, reduce costs, facilitate the sharing of other services. However, when the data is stored in the cloud storage device, a long time, enterprises and users inevitably will have security concerns, fearing that the information is actually stored in the cloud is still in the storage device or too long without access to, has long been the cloud server removed or destroyed, resulting in businesses and users in the future can't access or restore the data files. Therefore, this scheme goal to research and design fordata storage cloud computing environments that are proved. Stored in the cloud for data storage, research and develop a security and efficient storage of proof protocol, also can delegate or authorize others to public verifiability whether the data actually stored in the cloud storage devices

In 2008---IEEE Transactions—MR-PDP: Multiple-replica provable data possession,

Many storage systems rely on replication to increase the availability and durability of data on untrusted storage systems. At present, such storage systems provide no strong evidence that multiple copies of the data are actually stored. Storage servers can collude to make it look like they are storing many copies of the data, whereas in reality they only store a single copy. We address this shortcoming through multiple-replica provable data possession (MR-PDP): A provably-secure scheme that allows a client that stores t replicas of a file in a storage system to verify through a challenge-response protocol that (1) each unique replica can be produced at the time of the challenge and that (2) the storage system uses t times the storage required to store a single replica. MR-PDP extends previous work ondata possession proofs for a single copy of a file in a client/server storage system (Ateniese et al., 2007). Using MR-PDP to store t replicas is computationally much more efficient than using a single-replica PDP scheme to store t separate, unrelated files (e.g., by encrypting each file separately prior to storing it). Another advantage of MR-

PDP is that it can generate further replicas on demand, at little expense, when some of the existing replicas fail.

In 2009---IEEE Transactions: HAIL: A high-availability andintegrity layer for cloud storage:-The computing power in a cloud computing environments is supplied by a collection of data centers, orcloud data storages (CDSs) housed in many different locations and interconnected by high speed networks. CDS, like any other emerging technology, is experiencing growing pains. Data integritychecking of data and data structures has grown in importance recently in cloud computing due to the expansion of online cloud services, which have become reliable and scalable. In this paper we propose an integrity layered architecture of a typical cloud based on MAS architecture consists of two mainlayers cloud resources layer (cloud server-side) and MAS thecloud resources layer there exist massive physical cloud resources(storage serversand cloudapplication servers) that power the CDS. MAS's architecture layers architecture layer (cloud client-side). At consist of two agents: CloudService Provider Agent (CSPA and Cloud Data Integrity Backup Agent (CDIBA). This layered)architecture named as “CloudZone”. A prototype of our proposed “CloudZone” will be designed using Prometheus Methodology and implemented using the Java Agent Development Framework Security

In 2012---IEEE Transactions –Distributed data possession checking for securing multiple replicas in geographicallydispersed clouds:-Many cloud storage providers declare that they store multiple replicas of clients' data in order to preventdata loss. However, currently there is no guarantee that they actually spend storage for multiplereplicas. Recently a multiple-replica provable data possession (MR-PDP) protocol is proposed, which provides clients with the ability to check whether multiple replicas are really stored at the cloud storage servers. However, in MR-PDP, only private verifiability is achieved. In this paper, we proposeamultiplereplica remote data possession checking pro tocol which has public verifiability. The public verifiability increases the protocol's flexibility in that a third-party auditor can perform the data checkingon behalf of the clients. Homomorphic authentication tags based on BLS signature are used in the proposed protocol. By security analysis and performance analysis, the proposed protocol is shown to be secure and efficient, which makes it very suitable in cloud storage systems.

A .PRELIMINARIES AND PROBLEM STATEMENT

Regenerating Codes: Regenerating codes are first introducedlessen the repair data transfer capacity. Seeing distributedstorage to be an accumulation of n stockpiling servers, information record F is encoded what's more, put away repetitively over these servers. At that point F can be recovered by associating with any k-out-of-n servers, which is termed the MDS2-property. At the point when information debasement at a server is recognized, the customer will contact sound servers what's more, download β bits from every server, in this way recovering the ruined squares without recuperating the whole unique record. Dimakis et al. [18] demonstrated that the repair transmission capacity γ = β can be altogether decreased with ≥ k. Besides, they examined the crucial tradeoff between the capacity cost αand the repair transfer speed γ, then displayed two amazing and for all intents and purposes pertinent focuses on the ideal tradeoff bend: the base transfer speed recovering (MBR) point, which speaks to the working point with the slightest conceivable repair data transfer capacity, and the base stockpiling recovering (MSR) point,lightest conceivable

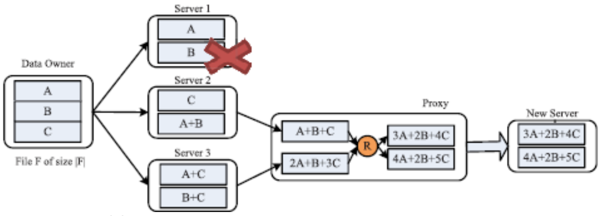

Fig. 1. *An example of functional repair regenerating code with parameters* $(n = 3,\ k = 2,\ \_ = 2,\ \alpha = 2,\ \beta = 1)$. The data owner computessix coded blocks as random linear combinations of the native three blocks, and distributes them across three servers. When Server 1 gets corrupted, the proxy contacts the remaining two servers and retrieves one block (obtained also by linear combination) from each, then it linearly combines them to generate two new coded blocks. Finally, the new coded blocks are sent to a new healthy server.

B.Model Framework

We consider the examining framework model for Regenerating- Code-based distributed storage as Fig.2, which includes four substances: the information proprietor,whopossesses a lot of information records to be put away in the cloud; the cloud, which are overseen by the cloud administration supplier, give stockpiling administration also,have noteworthy computational assets; the outsider evaluator(TPA), who has ability and capacities to direct open reviews on the coded information in the cloud, the TPA is trusted what's more, its review result is fair-minded for both information proprietors and cloud servers; and an intermediary specialist, who is semi-trusted and follows up on benefit of the information proprietortorecover authenticators and information hinders on the fizzled servers amid the repair technique. Notice that the information proprietor is confined in computational and capacity assets contrasted with different substances and may gets to be logged off indeed, even after the information transfer strategy. The intermediary, who might continuouslybe on the web, should be considerably more intense than the information proprietor however not exactly the cloud servers in wording of calculation and memory limit. To spare assets as well as the online weight conceivably brought by the intermittent reviewing and unintentional repairing, the information proprietors resort.

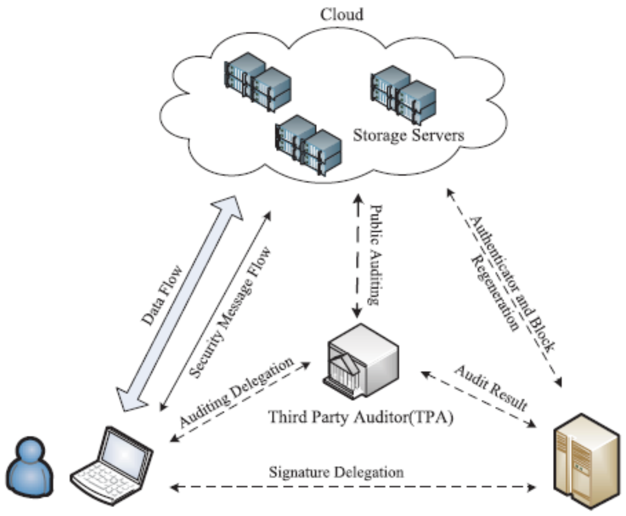

Fig-2 *The system model*

C.Meanings of Our Auditing Scheme
Our evaluating plan comprises of three methods:
Setup, Audit and Repair. Every strategy contains certain polynomial-time calculations as takes after

Setup: The information proprietor keeps up this methodology to introduce the examining plan.

KeyGen(1κ) → (pk, sk): This polynomial-time calculation is controlled by the information proprietor to instate its open and mystery parameters by taking a security parameter κ as information.

Degelation(sk) → (x): This calculation speaks to the connection between the information proprietor and intermediary. The information proprietor conveys halfway mystery key x to the intermediary through a protected approach.

SigAndBlockGen(sk, F) → This polynomial time calculation is controlled by the information proprietor and takes the mystery parameter sk and the first document F as information, and after that yields a coded piece set an authenticator set and a document label t.

Review: The cloud servers and TPA collaborate with each other to take an arbitrary example on the pieces and check the information soundness in this system.

Challenge(Fin f o) → (C): This calculation is performed by the TPA with the data of the record Fin f o as information and a challenge C as yield.

Verification Gen)→(P): This calculation is controlled by each cloud server with information challenge C, coded piece set what's more, authenticator set , then it yields a proof P.

Veri f y(P, pk, C)→ (0, 1): This calculation is controlled by TPA promptly after a proof is gotten. Taking the confirmation P, open parameter pk and the comparing challenge C asinformation, it yields 1 if the check passed and 0 generally.

Repair: without the information proprietor, the intermediary interfaces with the cloud servers amid this methodology torepair the wrong server identified by the inspecting processings of Our Auditing Scheme controlled by TPA.

ClaimForRep(Finfo) → (Cr): This calculation is comparable with the Challenge() calculation in the Audit stage, yet yields a case for repair Cr .

Gen For Rep(Cr,Φ,ψ) → (BA): The cloud servers run calculation after getting output the Cr lastly yield the square and authenticators set BA with another two inputs ,

BlockAndSigReGen(Cr, BA) → (Φ’,ψ’⅃): The proxy implements this algorithm with the claim Cr .

The sequence chart of the scheme is shown below in Fig.3

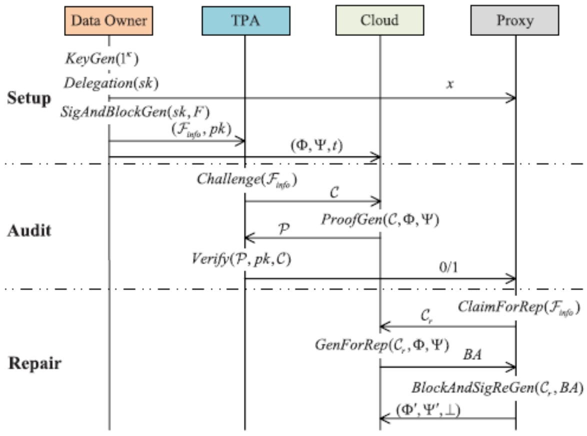
Data Owner
TPA
Cloud
Proxy
Setup
Audit
Repair
$KeyGen(1^\kappa)$
$Delegation(sk)$
x
$SigAndBlockGen(sk,F)$
$(\mathcal{F}_{info}, pk)$
(Φ,Ψ,t)
$Challenge(\mathcal{F}_{info})$
$\mathcal{C}$
$ProofGen(\mathcal{C},\Phi,\Psi)$
$\mathcal{P}$
$Verify(\mathcal{P},pk,\mathcal{C})$
0/1
$ClaimForRep(\mathcal{F}_{info})$
$\mathcal{C}_r$
$GenForRep(\mathcal{C}_r,\Phi,\Psi)$
BA
$BlockAndSigReGen(\mathcal{C}_r,BA)$
$(\Phi',\Psi',\perp)$

Fig .3 *The sequence chart of our scheme*

II. THE PROPOSED SYSTEM

In this plan a novel authenticator, which is more proper for recovering codes. In addition, we "encode" the coefficients to secure information protection against the inspector, which is more lightweight than applying the confirmation blind strategy in and information blind technique in . A few difficulties and dangers suddenly emerge in our new framework model with an intermediary and security investigation demonstrates that our plan strategy with comprehend this i Fig.5 analyzes the running time of setup stage using three distinctive authenticator era techniques (with variable s = 60, 80, 100): the direct approach, our primitive approach and our appointment approach Clearly, all the three approaches present higher time cost with bigger s, as there are more units need to sign; all things considered, our own is continuously more proficient than the direct approach. The reason is that our primitive era technique diminishes the times of the tedious measured example operations to nαs(m+1) + 2m contrasted with the clear approach,which requires nαs(m + 3) operations completely. In any case, it is still escalated asset expending and even unreasonably expensive at the point when information proprietors use asset constrained hardware (e.g., Tablet PC et al.) to sign their information pieces and transfer them. In this manner, we acquaint another appointment ssue. Exploratory result demonstrates that our appointment technique discharges the information proprietor from overwhelming computational overhead; the computational many-sided quality of information proprietor is diminished to around 1/18 of that with our primitive strategy.

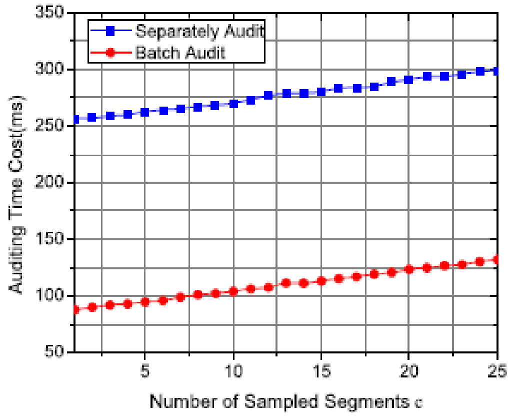

Fig .4Time of audit with different c.

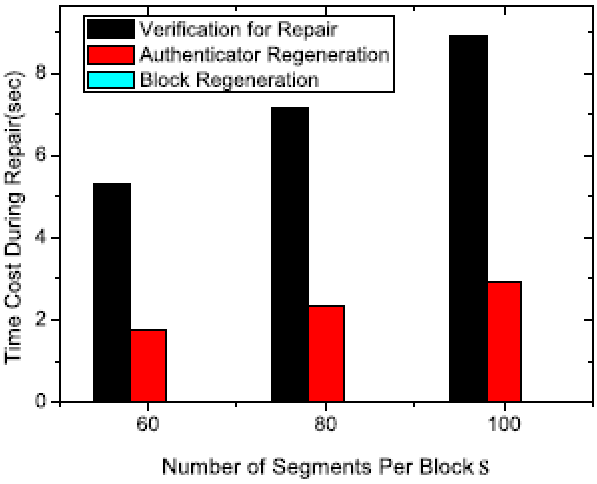

Fig .5.Time of repair with different s

III. CONCLUSION

In this paper, we propose an open reviewing plan for the recovering code-based distributed storage framework, where the information proprietors are advantaged to assign TPA for their information legitimacy checking. To secure the firstinformation protection against the TPA, we randomize the coefficients at the outset rather thaapplying the visually impaired method amid the reviewing process. Considering that the information proprietor can't generally stay online in rehearse, so as to keep the capacity accessible and irrefutable after a malignant defilement, we present a semi-trusted intermediary into the framework demonstrate and give a benefit to the intermediary to handle the reparation of the coded squares and authenticators. To better suitable for the reTo better appropriate for the regenerating-code-scenario,we design our authenticator based on the BLS signature.Thisauthenticator can be efficiently generated by thedataowner simultaneously with the encoding procedure. Extensiveanalysis shows that our scheme is provable secure, and the performanceevaluation shows that our scheme is highlyefficientand can be feasibly integrated into a regenerating-code-based.cloud storage systemcovering code-situation, we outline our authenticato*r in view of the BLS* signature

$$
\begin{aligned}
e(\sigma_i, g_2) &= e\left(\prod_{j=1}^{\alpha} \sigma_{ij}^{a_j}, g_2\right) \\
&= \prod_{j=1}^{\alpha} e\left(\sigma_{ij}, g_2\right)^{a_j} \\
&= \prod_{j=1}^{\alpha} e\left(\prod_{\tau=1}^{c} \sigma_{ijk_\tau^*}^{a_\tau^*}, g_2\right)^{a_j} \\
&= \prod_{j=1}^{\alpha}\left(e\left(\left(\prod_{\tau=1}^{c} H_{j\tau}^{a_\tau^*}\right)^{x}, g_2\right)\right)^{a_j} \\
&\quad \cdot \prod_{j=1}^{\alpha} e\left(\left(u^{\sum_{\tau=1}^{c} a_\tau^* v_{ijk_\tau^*}}\right)^{y}, g_2\right)^{a_j} \\
&\quad \cdot \prod_{j=1}^{\alpha} e\left(\left(\prod_{\lambda=1}^{m} w_\lambda^{\varepsilon_{ij\lambda}}\right)^{y \sum_{\tau=1}^{c} a_\tau^*}, g_2\right)^{a_j} \\
&= \prod_{j=1}^{\alpha}\left(e\left(\prod_{\tau=1}^{c} H_{j\tau}^{a_\tau^*}, pk_x\right)\right)^{a_j} \\
&\quad \cdot \prod_{j=1}^{\alpha} e\left(u^{\mu_{ij}}, pk_y\right)^{a_j}
\end{aligned}
$$

Given $g, g^x, g^y \in G$, the algorithm $\mathcal{S}$ maintains the chal- nger in Game 0 and interacts with adversary $\mathcal{A}$ as follows: *Setup:* $\mathcal{S}$ runs the KeyGen() algorithm on input 1^κ and gives : public parameter (pk_x, pk_y) to adversary $\mathcal{A}$.
Queries: $\mathcal{S}$ operates the oracles as follows.

- *Hash Oracle($\mathcal{O}_{hash}$):* For each query with input tuple $\{ID, i, j, k\}$, if it has already been queried, the oracle returns H_{ijk}. If this query is a new one, $\mathcal{S}$ guesses whether this current query is for forgery or not, if so, it sets $H_{ijk} = g^y$, otherwise sets $H_{ijk} = g^{\gamma_{ijk}}$ with $\gamma_{ijk} \overset{R}{\leftarrow} GF(p)$.
- *UW Oracle($\mathcal{O}_{uw}$):* For each query for u, $w_\lambda(1 \leq \lambda \leq m)$ with input $\{ID, i, j, k, v_{ijk}, \{\varepsilon_{ij\lambda}\}_{\lambda=1}^{m}\}$, if this tuple has been queried, returns corresponding u, $w_\lambda(1 \leq \lambda \leq m)$. Otherwise, $\mathcal{S}$ randomly chooses $r, r_\lambda \in GF(p), 1 \leq \lambda \leq m$, and guesses whether this current query is for forgery or not, if so, $\mathcal{S}$ sets

thus we get

$$
e\left(\prod_{\lambda=1}^{m} w_\lambda^{-\rho_{i\lambda}}, pk_y\right) \cdot e\left(\sigma_i, g_2\right) = RHS.
$$

$u = (g^x)^r$, $w_\lambda = (g^x)^{r_\lambda}(1 \leq \lambda \leq m)$, if not,

$u = (g^\zeta)^r$, $w_\lambda = (g^\zeta)^{r_\lambda}(1 \leq \lambda \leq m)$ with $\zeta \overset{R}{\leftarrow} GI$

- *Sign Oracle*($\mathcal{O}_{sign}$): For each input $\{pk_x, pk_y\}$ and tuple $\{ID, i, j, k, \upsilon_{ijk}, \{\varepsilon_{ij\lambda}\}_{\lambda=1}^{m}\}$, $\mathcal{S}$ outputs the $\sigma_{ijk} = (g^{\gamma_{ijk}})^x((g^{\zeta})^{r\upsilon_{ijk}+\sum_{\lambda=0}^{m} r_\lambda \varepsilon_{ij\lambda}})^y$ by querying the $\mathcal{O}_{hash}$ and $\mathcal{O}_{uw}$.

Output: Eventually, the adversary $\mathcal{A}$ outputs a result tuple $\{ID^*, i,^* j^*, k^*, \upsilon^*, \{\varepsilon_\lambda^*\}_{\lambda=1}^{m}, \sigma^*\}$, if Eq.(23) holds while the input tuple $\{ID^*, i,^* j^*, k^*, \upsilon^*, \{\varepsilon_\lambda^*\}_{\lambda=1}^{m}\}$ was not submitted to the Sign Oracle $\mathcal{O}_{sign}$, $\mathcal{A}$ wins Game 0 and $\mathcal{S}$ declares "*success*", otherwise $\mathcal{A}$ loses Game 0 and $\mathcal{S}$ aborts.

If $\mathcal{S}$ declares "*success*", she obtains $\sigma^* = (g^y)^x((g^x)^{r\upsilon^*+\sum_{\lambda=1}^{m} r_\lambda \varepsilon_\lambda^*})^y = (g^{xy})^{1+r\upsilon^*+\sum_{\lambda=0}^{m} r_\lambda \varepsilon_\lambda^*}$. Thus the proposed CDH problem can be solved as $g^{xy} = (\sigma^*)^{\frac{1}{1+r\upsilon^*+\sum_{\lambda=0}^{m} r_\lambda \varepsilon_\lambda^*}}$.

The probability that $\mathcal{S}$ guesses the forged input tuple is more than $1/q_h$, then $\mathcal{S}$ can solve the CDH problem with probability $\epsilon' \geq \epsilon/q_h$ which is non-negligible. □

REFERENCES

[1] M. Armbrustet *al.*, "Above the clouds: A Berkeley view of cloud computing," Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, CA, USA, Tech. Rep. UCB/EECS-2009-28, 2009.

[2] G. Atenieseet *al.*, "Provable data possession at untrusted stores," in *Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, 2007, pp. 598–609.

[3] A. Juels and B. S. Kaliski, Jr., "PORs: Proofs of retrievability for large files," in *Proc. 14th ACM Conf. Comput. Commun. Secur.*, 2007, pp. 584–597.

[4] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PDP: Multiple-replica provable data possession," in *Proc. 28th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2008, pp. 411–420.

[5] K. D. Bowers, A. Juels, and A. Oprea, "HAIL: A high-availability and integrity layer for cloud storage," in *Proc. 16th ACM Conf. Comput. Commun. Secur.*, 2009, pp. 187–198.

[6] J. He, Y. Zhang, G. Huang, Y. Shi, and J. Cao, "Distributed data possession checking for securing multiple replicas in geographicallydispersed clouds," *J. Comput. Syst. Sci.*, vol. 78, no. 5, pp. 1345–1358, 2012.

[7] B. Chen, R. Curtmola, G. Ateniese, and R. Burns, "Remote data checking for network coding-based distributed storage systems," in *Proc. ACM Workshop Cloud Comput. Secur. Workshop*, 2010, pp. 31–42.

[8] H. C. H. Chen and P. P. C. Lee, "Enabling data integrity protection in regenerating-coding-based cloud storage: Theory and implementation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 2, pp. 407–416, Feb. 2014.

[9] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 9, pp. 1717–1726, Sep. 2013.

[10] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 12, pp. 2231–2244, Dec. 2012.

[11] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," *Proc. IEEE*, vol. 99, no. 3, pp. 476–489, Mar. 2011.

[12] H. Shacham and B. Waters, "Compact proofs of retrievability," in *Advances in Cryptology*. Berlin, Germany: Springer-Verlag, 2008, pp. 90–107.

[13] Y. Hu, H. C. H. Chen, P. P. C. Lee, and Y. Tang, "NCCloud: Applying network coding for the storage repair in a cloud-of-clouds," in *Proc. USENIX FAST*, 2012, p. 21.

[14] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[15] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Trans. Comput.*, vol. 62, no. 2, pp. 362–375, Feb. 2013.

[16] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," *IEEE Trans. Service Comput.*, vol. 5, no. 2, pp. 220–232, Apr./Jun. 2012.

[17] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," *J. Cryptol.*, vol. 17, no. 4, pp. 297–319, 2004.

[18] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. J. Wainwright

[19] T. Ho *et al.*, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.

[20] D. Boneh, D. Freeman, J. Katz, and B. Waters, "Signing a linear subspace: Signature schemes for network coding," in *Public Key Cryptography*. Berlin, Germany: Springer-Verlag, 2009, pp. 68–87.

[21] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology*. Berlin, Germany: Springer-Verlag, 2001, pp. 213–229.

[22] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for FR-reduction," *IEICE Trans. Fundam. Electron., Commun.,Comput. Sci.*, vol. E84-A, no. 5, pp. 1234–1243, 2001.

[23] R. Gennaro, J. Katz, H. Krawczyk, and T. Rabin, "Secure network coding over the integers," in *Public Key Cryptography*. Berlin, Germany: Springer-Verlag, 2010, pp. 142–160.

[24] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM J. Comput.*, vol. 17, no. 2, pp. 281–308, 1988.

[25] P. S. L. M. Barreto and M. Naehrig, "Pairing-friendly elliptic curves of prime order," in *Selected Areas in Cryptography*. Berlin, Germany: Springer-Verlag, 2006, pp. 319–331.

[26] Y. Deswarte, J.-J. Quisquater, and A. Saïdane, "Remote integrity checking," in *Integrity and Internal Control in Information Systems VI*. Berlin, Germany: Springer-Verlag, 2004, pp. 1–11.

[27] D. L. G. Filho and P. S. L. M. Barreto, "Demonstrating data possession anduncheatable data transfer," Cryptology ePrint Archive, Tech. Rep. 2006/150, 2006. [Online]. Available: http://eprint.ia