A Protected Removal Code-Based Cloud Storage Space Scheme with Protected Information Transferring

R. Priyadarshini[1], P. Karthikeyan[2], A. Dhasaradhi[3], T. Thiresa[4]
[1,4]M.Tech Student, [2,3]Assistant Professor
[1,3]Sidharth Institute of Engineering & Technology
[2]Lord Venkateshwaraa Engineering College
[4]Seshachala Institute of Technology

***Abstract*—** A cloud storage space scheme, consisting of a set of storage space servers, provides long-term storage space services more than the Internet. Storing information in a third party's cloud system cause grave worry more than information privacy. Universal encryption scheme defend information privacy, but also bound the functionality of the storage space system because a few operations are support more than encrypted information. Build a protected storage space scheme that supports various functions is challenging when the storage space scheme is spread and has no central authority. We suggest a doorsill proxy re-encryption system and merge it with a decentralized erasure policy such that a protected distributed storage space scheme is formulated. The distributed storage space scheme not only supports protected and robust information storage space and recovery, but as well lets a user promote his information in the storage servers to another user without retrieve the information back. The major technical role is that the proxy re-encryption system supports encoding operations more than encrypted communication as well as forwarding operations over encoded and encrypted communication. Our scheme completely integrates encrypting, encoding, and forwarding. We evaluate and propose appropriate parameter for the number of copies of a message dispatch to storage space servers and the number of storage space servers queried by a key server. These parameters permit additional flexible tuning between the number of storage space servers and robustness.

Keywords-; storage space; proxy re-encryption system; spread storage ; third party .

I. INTRODUCTION

As speedy networks and omnipresent Internet access develop into on hand in recent years, a lot of services are provide on the Internet such that users can utilize them from anyplace at any moment. For example, the email service is most likely the most accepted one. Cloud computing is a copy that treat the possessions on the Internet as a integrated thing, a cloud. Users just utilize services without being disturbed about how working out is done and storage space is manage. In this paper, we concentrate on designing a cloud storage space scheme for robustness, secrecy, and functionality. A cloud storage space scheme is consider as a big range spread storage space scheme that consists of a lot of independent storage servers.

Data robustness is a most important necessity for storage space scheme. There enclose a lot of proposals of storing information more storage space servers. Single approach to give information strength is to duplicate a message such that each storage space server stores a duplicate of the significance It is extremely robust for the reason that the message can be retrieve as extended as one storage space server survive. One more method is to encode a message of k secret language into a secret word of n secret code by removal code. To accumulate a letter, every of its secret word cipher is store in a dissimilar storage space server. A storage space server crash corresponds to an removal error of the secret word character. As extended as the quantity of crash servers is below the acceptance entrance of the removal policy, the data can be recovered starting the secret word symbols store in the obtainable storage space servers by the decode method. These provide a exchange between the storage space volume and the acceptance entrance of crash servers.. Therefore, the training procedures for a letter can be opening into n similar responsibilities of generate secret word secret language. A decentralized removal policy is appropriate for utilize in a spread storage space scheme. Later than the data secret language are send to storage space servers, each storage space server separately compute a secret word character for the conventional data secret code and store it. This finish the encoding and store procedure. The revival procedure is the similar.

Store information in a third party's cloud scheme cause grave anxiety on information privacy. In arrange to offer strong secrecy for data in storage space servers, a client can encrypt data by a cryptographic technique earlier than apply an removal policy technique to code and store data. When client needs to utilize a data, client wants to recover the secret word signs from storage space servers, decipher them, and then decrypt them through use cryptographic key. Here are three troubles in the more than frank combination of encryption and encoding.

Primary, the client has to act mainly calculation and the data transfer among the client and storage space servers is elevated. Second, the clients have to control his cryptographic key. If the user's machine of store the keys are missing or compromise, the protection is busted. Lastly, as well information store and retrieve, it is firm for storage space servers to straight maintain extra function. For example, storage space servers cannot straight forward a client data to a new one. The vendors of data have to recover, make out, decrypt and next onward them to a new client. In this paper, we speak to the difficulty of forward information to a new client by storage space servers straight below the control of the information vendor. We believe the method representation to consist of spread storage space servers and type servers. Since a store cryptographic key in a solo device is dangerous, a client distributes his cryptographic key to key servers that shall execute cryptographic function on behalf of the client. These key servers are extremely secluded by safety mechanism. To robust the spread constitution of system, we need to servers separately do all operation. By this reflection, we recommend a new threshold proxy re-encryption method and incorporate it among a protected decentralized policy to form a safe spread storage space scheme.

The encryption method supports encoding operation more than encrypted data and forward operation more than encrypted and fixed data. The stiff addition of encoding, encryption, and forward make the storage space scheme powerfully gather the necessities of information strength, information privacy, and information forward. Accomplishing the addition among kindness of a spread arrangement is tough. Our scheme meets the necessities that storage space servers separately do encoding, re-encryption and key servers separately do incomplete decryption. Furthermore, we believe the scheme in a additional universal surroundings than before mechanism. These sets allow additional elastic modification among the quantity of storage space servers and strength.

II. RELATED WORKS

We briefly analysis spread storage space scheme, proxy re-encryption method, and reliability inspection method.

A. spread storage space scheme

On the early days, the Network-Attached Storage (NAS) and the Network File System (NFS) offer more Storage space policy more than the network such to facilitate a client can access the storage space procedure via system connection. Later, a lot of improvement on scalability, strength, effectiveness, and safety be designed

A decentralized design for storage space scheme offers excellent scalability; because a storage space server can connect or run off exclusive of organize of a mid ability. To offer strength in opposition to server breakdown, a easy technique is to build duplication of every data and accumulate them in various servers. However, this technique is costly as z duplication effect in z time of spreading out.

One technique to decrease the growth rate is to utilize removal code to coded data. A data is set as a codeword, which is a vector of signs, and every storage space server supplies a secret word symbol. A storage space server breakdown is model as a removal fault of the store secret word representation. To accumulate a data of k block, every storage space server linearly merges the blocks with arbitrarily selected coefficients and supplies the secret word representation and coefficients. To recover the data, a client query k storage space servers for the accumulate secret word signs and coefficients and explain the linear method. The techniques have light data privacy for the reason that invaders can cooperation k storage space servers to obtain the data.

*B. **Proxy Re-Encryption methods***

Proxy re-encryption methods are proposed by Mambo and Okamoto and Blaze et al. In a proxy re-encryption method, a proxy server can move a code text below a public key PKA to a fresh one below one more public key PKB by use the re-encryption key RKA!B. The servers do not identify the plaintext through change. Ateniese et al projected a few proxy re-encryption ideas and apply them to the distribution task of protected storage space scheme. In their effort, data's are primary encrypted by the vendor and next accumulate in a storage space server. When a client needs to distribute his data, he sends a re-encryption key to the storage space server. The storage space server re-encrypts the encrypted data for the approved client. Thus, their scheme has information privacy and supports the information forward task. Our effort more combines encryption, re-encryption, and encoding such that storage space strength is strengthening.

Type-based proxy re-encryption method projected by Tang offer a enhanced granularity on the arranged exact of are-encryption key. A client can choose which kind of data and with whom he needs to contribute to in this type of proxy re-encryption technique Key-private proxy re-encryption technique are proposed by Ateniese et al. . In a key-private proxy re-encryption method, known a re-encryption key, a proxy server cannot find out the identity of the receiver. This type of proxy re-encryption method offer advanced confidentiality security aligned with proxy servers. While the majority proxy re-encryption methods utilize combination function, here survive proxy re-encryption method lacking combination

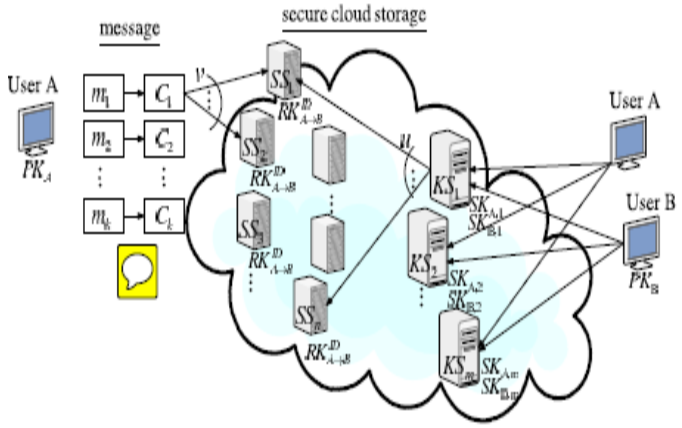

Fig. 1. A universal scheme representation of our effort

C. Reliability Inspection Functionality

A new main functionality regarding cloud storage space is the task of reliability inspection. Later than client supplies information into the storage space scheme, his rejection longer possesses the information at give. The client might desire to verify whether the information are correctly store in storage space servers. The idea of verifiable information control and the idea of evidence of storage space are projected. Later on, unrestricted check capability of store information is addressed in. However every one of them believes the data in the plain text type.

III. DEVELOPMENT

We here the development of the storage space scheme, the hazard replica that we believe for the privacy matter, and a conversation for a simple answer

A. System Representation

Our scheme representation consists of clients, n storage space servers SS1; SS2; . . . ; SSn, and m key servers KS1; KS2; . . . ; KSm. storage space servers offer storage space service and key servers offer key running services. They effort separately. Our spread storage space scheme consists of four phases: scheme arrangement, information storage space, information forwarding, and information recovery. These four phases are express as follows.

In the scheme arrangement part, the scheme manager selects scheme parameter and distributes them. All clients A is allocate a public-secret key pair (PKA; SKA). Client A distributes his secret key SKA to key servers such to every key server KSi embrace a key split SKA. The key is common with an entrance t.

In the information storage space period, client A encrypts his data M and sends out it to storage space servers. A data M is decayed into k block m1;m2; . . .;mk and have an identifier ID. client A encrypts every block mi into a coded text Ci and send it to v arbitrarily selected storage space servers. Ahead getting coded texts from a client, every storage space server linearly merge them with arbitrarily selected coefficients into a secret word representation and store it. Note down that a storage space server might collect a lesser amount of than k data's block and we guess that all storage space servers make out the value k in advance.

In the information forward stage, client A forward his encrypted data with an identifier ID store in storage space servers to client B such that can decrypt the forward data by his secret key. To do so, A use his top secret type SKA and B's free key PKB to work out a re-encryption key RKID A!B and then send RKID A!B to all storage space servers. Every storage space server use the re-encryption key to re-encrypt its secret word sign for later on recovery desires by B. The re-encrypted secret word sign is the grouping of cipher texts less than B's public key. In arrange to differentiate re-encrypted secret word signs from whole ones, we identify them innovative secret word signs and re-encrypted secret word signs, respectively

In the information recovery period, client A needs to recover a data from storage space servers. The data is also store in and forward to him. Client A send a recovery call to key servers. Ahead in receipt of the recovery call and execute a correct verification procedure with client A, every key server KSi needs u arbitrarily selected storage space servers to obtain secret word signs and do incomplete decryption on the conventional secret word signs by use the key split SKA.i. At last, client A combine the incompletely decrypted secret word signs to get the unique data M

Scheme improving. after a storage space server stop working, a fresh one is added. The fresh storage space server query k accessible storage space servers linearly combine the established secret word signs as a fresh one and stores it. The scheme is then improved.

We believe information privacy for both information storage space and information forward. In this warning replica, an enemy wishes to break information privacy of a aim client. To perform so, the enemy colludes with all servers, non-target user, and up to (t-1) key servers. The enemy analyzes store data in storage space servers, the surreptitious keys of non plan clients and the common keys store in key servers. Note that the storage space servers accumulate all re-encryption key provide by clients. The enemy may try to produce a innovative re-encryption key from store re-encryption keys. We properly representation this attack by the normal chosen plaintext attack1 of the alternate.

A cloud storage space scheme model in the exceeding is protected if no probabilistic polynomial instant enemies be successful the entertainment with a non insignificant advantage. A protected cloud storage space scheme implies to an illegal client or server cannot obtain the satisfied of accumulate data, and a storage space server cannot produce re-encryption keys by himself. If a storage space attendant can make a re-encryption key beginning the goal client to one more client B, the enemy can succeed the safety entertainment by re-encrypting the secret message text to B and

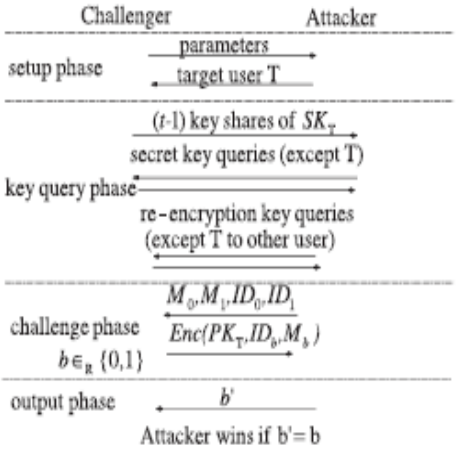

Fig. 2. The protection game for the selected plaintext harass

decrypting the re-encrypted cipher text by the covert key SKB. So, this replica addresses the safety of data storage space and information forwarding.

A cloud storage space scheme model in the beyond is safe if no probabilistic polynomial instance enemy win the amusement with a non insignificant benefit. A safe cloud storage space scheme implies that an illegal client or server cannot get the comfortable of store data, and a storage space server cannot produce re-encryption keys by himself. If a storage space server can produce a re-encryption key from the aim client to one more client B, the enemy can win the protection game by re-encrypting the secret message text to B and decrypting the re-encrypted secret message text by the secret key SKB. So, this replica addresses the safety of information storage space and information forwarding.

B. A Simple Clarification

A simple clarification to behind the information forward function in a spread storage space scheme is as follow: As the vendor A needs to ahead a data to client B, he downloads the encrypted data and decrypts it by use his secret key. He after that encrypts the data by use B's open key and uploads the fresh cipher text. When B needs to recover the forward data from A, he downloads the cipher text and decrypts it with his secret key. The entire information forward procedure wants three statement rounds for A's downloading and uploading and B's downloading. The announcement cost is linear in the extent of the forward data. The working out cost is the decryption and encryption for the possessor A, and the decryption for client B.

Proxy re-encryption method can extensively reduce announcement and calculation cost of the vendor. In a proxy re-encryption method, the owner send a re-encryption key to storage space servers such that storage space servers perform the re encryption process for him. Thus, the announcement cost of the possessor is self-determining of the duration of forwarded data and the calculation cost of re-encryption is in use heed of by storage space servers. Proxy

re-encryption method significantly reduces the above your head of the information forward utility in a protected storage space scheme

IV. BUILD FOR PROTECTED CLOUD STORAGE SPACE SCHEME

Earlier than present our storage space scheme, we briefly begin the arithmetical situation, the stability hypothesis, an removal policy over exponent, and our approach. Bilinear map. Let GG1 and GG2 be repeated multiplicative group with a major sort p and g G€G1 be a producer

Our approach. We exercise a threshold proxy re-encryption method with multiplicative homomorphism possessions. An encryption method is multiplicative homomorphism if it supports a set procedure _ on encrypted plaintexts exclusive of decryption

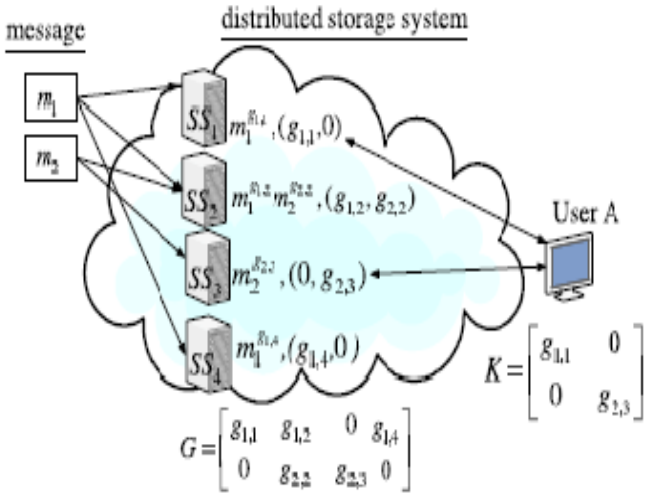

Fig. 3. A storage space scheme with arbitrary linear coding more than exponents

D (S K, E (P K, m1) Θ E (P K, m2)) = m1, m2

Wherever E is the encryption task, D is the decryption task, and (PK;SK) is a couple of public key and secret key. Specified two coefficients g1 and g2, two data signs m1 and m2 can be fixed to a secret word sign mg1 mg2 2 is the encrypted shape Therefore, multiplicative homomorphism encryption methods supports the encoding process more than encrypted data's .We then change a proxy re-encryption method with multiplicative homomorphism assets into a entry version. A secret key is common to key servers with a threshold cost t via the Shamir underground distribution method, where t >= k. In our scheme, to decrypt for a set of k data signs, each key server separately queries 2 storage space servers and incompletely decrypts two encrypted secret word signs. As long as t key servers are obtainable, k secret word symbols are getting from the incompletely decrypted coded texts

A. A protected cloud storage space scheme with safe forwarding

Scheme arrangement. The algorithm arrangement generate the scheme parameter .A client uses KeyGen(P) to produce his open and furtive key pair and ShareKeyGe(p)to split his furtive key to a situate of m key servers with a doorsill t, where k _ t _ m. The client nearby stores the third constituent of his furtive key A, every key server KSi needs u arbitrarily selected storage space servers to obtain secret word signs and do incomplete decryption on the conventional secret word signs by use the key split SKA.i. The encryption method supports encoding operation more than encrypted data and forward operation more than encrypted and fixed data. The stiff addition of encoding, encryption, and forward make the storage space scheme powerfully gather the necessities of information strength, information privacy, and information forward. Accomplishing the addition among kindness of a spread arrangement is tough

Information Storage Space. When client A requirements to stock up a significance of k block m1;m2; . . .;mk with the identifier ID, he calculates the individuality sign and achieve the encryption algorithm and k block to get k inventive secret message texts C1; C2; . . . ; Ck. An imaginative secret message text is indicate by a leading bit b. client A send each secret message text Ci to v arbitrarily selected storage space servers. A storage space server

receives a situate of inventive secret message texts with the similar identity symbol from A. When a secret message text Ci is not established, the storage space server insert Ci to the set. The particular arrangement of is a stain for the nonappearance of Ci. The storage space server performs Encode on the set of k secret message texts and stores the fixed result (secret word sign).

$$\begin{aligned} C' &= \left(0, \prod_{i=1}^{k}(\alpha_i^{g_i}), \beta, \prod_{i=1}^{k}(\gamma_i^{g_i})\right) \\ &= \left(0, g^{\sum_{i=1}^{k} g_i r_i}, \tau, \prod_{i=1}^{k} m_i^{g_i} \tilde{e}(g^{a_1}, \tau)^{\sum_{i=1}^{k} g_i r_i}\right) \\ &= (0, g^{r'}, \tau, W\tilde{e}(g, \tau)^{a_1 r'}), \end{aligned}$$

MessageTransfering. Client A needs to transfer data to one more client B. He wants the first component a1 of his furtive key. If A do not acquire a1, he query key servers for key share. As soon as at slightest t key servers react, A recover the first module a1 of the furtive key SKA via the KeyRecover(.) algorithm. Let the identifier of the data be ID. Consumer A compute the re-encryption key RKID A!B via the ReKeyGen(.) algorithm and strongly sends the re-encryption key to each storage space server. By by means of RKID A!B, a storage space server re-encrypts the innovative secret word sign C0 with the identifier ID into a re-encrypted secret word sign C00 via the ReEnc(.)Algorithm such that C0 is decrypt able by use B's furtive key. A re-encrypted secret word sign is indicated by the leading bit . Let the community key PKB of client B

Information Recovery. Here are two cases for the information recovery stage. The first case is that a client A retrieve his have data. When client a needs to recover the data with the identifier ID, he inform every key servers through the individuality token .A key server primary retrieve unusual secret word signs from u arbitrarily selected storage space servers and then perform incomplete decryption. The key servers send the incompletely decrypted secret word signs and the coefficients to client A. later than devotee A accumulate rejoin beginning at least t key servers and at least k of them are initially from separate storage space servers, he execute on the t incompletely decrypted secret word signs to improve the blocks m1;m2; . . .;mk. The second case is that a user B retrieves a data forward to him. User B inform all key servers straight. The group and combine parts are the same as the first case except for that key servers recover re-encrypted secret word symbols and do incomplete decryption Share-on re-encrypted secret word signs

B. Analysis

We evaluate storage space and calculation complexity, accuracy, and protection of our cloud storage space scheme in this part. Let the bit-length of an part in the collection G1 be l1 and G2be l2. Let coefficients gij be arbitrarily selected from (0,1)

Storage cost. To store a data of k block, a storage space server SSj stores a codeword symbol (b,αj,t,rj) and the coefficient vector (g1j, g2j . . . ; gkjp). They are entirety of bit, anywhere. The standard price for a data bit store in a storage space server is (αij,t,w) bits, which is conquered by for a satisfactorily great k. In perform, small coefficients, i.e.,l3 l2, reduce the storage cost in each storage server. Computation cost. We measure the calculation price by the amount of coupling operation, modular exponentiation in GG1 and GG2, modular multiplications in GG1 and GG2, and arithmetic operations over . These operations are denoted as Pairing, Exp1, Exp2, Mult1, Mult2, and Fp, correspondingly. The cost is summarize in Table 1. compute an Fp takes a large amount less time than compute a Mult1 or a Mult2. The time of compute an Exp1 is 1:5dlog pe period as a lot as the point of compute a Mult1, on typical, (by using the square-and-multiply algorithm). Likewise, the time of compute a Exp2 is 1:5dlog pe period as a lot as the occasion of compute a Mult2, on regular.

Rightness. Here are two cases for rightness. The vendor A properly retrieves his data and client B properly retrieves a data forward to him. The rightness of encryption and decryption for A can be seen in . The rightness of re-encryption and decryption for B know how to be see in (2). As extended as at smallest amount k storage space servers are accessible,
a client can recover information with an irresistible possibility. Thus, our storage space scheme tolerates n _ k server failure.

The possibility of a flourishing recovery. A flourishing recovery is an occurrence that a client beneficially retrieves all k blocks of a data no subject whether the data is own by him or forward to him. The unpredictability come from the accidental collection of storage space servers in the information storage space stage, the accidental coefficients selected by storage space servers, and the accidental collection of key servers in the information recovery stage. The prospect of a flourishing recovery depends on (n; k; u; v) and all arbitrariness.

TABLE 1
The Computation Cost of Each Algorithm
in Our Secure Cloud Storage System

Operation	Computation cost
Enc	k Pairing + k Exp_1 + k Mult_2
Encode (for each storage server)	k Exp_1 + k Exp_2 + $(k-1)$ Mult_1 + $(k-1)$ Mult_2
KeyRecover	$O(t^2)$ F_p
ReKeyGen	1 Exp_1
ReEnc (for each storage server)	1 Pairing +1 Mult_2
ShareDec (for t key servers)	t Exp_1
Combine	k Pairing + t Mult_1 + $(t-1)$ Exp_1+$O(t^2+k^3)$ F_p + k^2 Exp_2 + $(k+1)k$ Mult_2

- Pairing: a pairing computation of $\tilde{e}$.
- Exp_1 and Exp_2: a modular exponentiation computation in $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively.
- Mult_1 and Mult_2: a modular multiplication computation in $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively.
- F_p: an arithmetic operation in $GF(p)$.

The method of investigation is parallel to that in and [6]. Though, we believe a dissimilar scheme replica from the one in and a more stretchy limitation situation for n =akc than the setting in and [6]. The differentiation among our scheme replica and the one in is that our scheme replica has key servers. In a solo client query k different storage space servers to recover the information On the additional hand, every key server in our scheme separately query u storage space servers. The use of spread key servers increases the stage of key security but makes the investigation harder.

V CONCLUSSIONS

In this paper, we believe a cloud storage space scheme consists of storage space servers and key servers. We add in a newly planned threshold proxy re-encryption method and removal code above exponent. The threshold proxy re-encryption method supports encoding, forward, and incomplete decryption operation in a spread way. Toward decrypt a communication of k blocks so as to be encrypted and set to n secret word symbols, every key server simply have to incompletely decrypt two secret word signs in our scheme. By use the threshold proxy re-encryption method, we present a protected cloud storage space scheme to provide safe information storage space and protected information forward functionality in a decentralized configuration. Furthermore, every storage space server separately performs encoding and re-encryption and every key server separately perform incomplete decryption.

Our storage space scheme and a little recently future satisfied addressable file system and storage space scheme are extremely companionable. Our storage space servers work as storage space nodes in a content addressable storage space scheme for store substance addressable block. Our key servers work as rights to use nodes for provide a front-end level such as a conventional file scheme edge. More learn on comprehensive collaboration is required.

REFERENCES

[1] J. Kubiatowicz, D. Bindel, Y. Chen, P. Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao, "Oceanstore: An Architecture for Global-Scale Persistent Storage," Proc. Ninth Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS), pp. 190- 201, 2000.

[2] P. Druschel and A. Rowstron, "PAST: A Large-Scale, Persistent Peer-to-Peer Storage Utility," Proc. Eighth Workshop Hot Topics in Operating System (HotOS VIII), pp. 75-80, 2001.

[3] A. Adya, W.J. Bolosky, M. Castro, G. Cermak, R. Chaiken, J.R.Douceur, J. Howell, J.R. Lorch, M. Theimer, and R. Wattenhofer, "Farsite: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment," Proc. Fifth Symp. Operating System Design and Implementation (OSDI),

[4] A. Haeberlen, A. Mislove, and P. Druschel, "Glacier: Highly Durable, Decentralized Storage Despite Massive Correlated Failures," Proc. Second Symp. Networked Systems Design and Implementation (NSDI), pp. 143-158, 2005.

[5] Z. Wilcox-O'Hearn and B. Warner, "Tahoe: The Least-Authority Filesystem," Proc. Fourth ACM Int'l Workshop Storage Security and Survivability (StorageSS), pp. 21-26, 2008.

[6] H.-Y. Lin and W.-G. Tzeng, "A Secure Decentralized Erasure Code for Distributed Network Storage," IEEE Trans. Parallel and Distributed Systems, vol. 21, no. 11, pp. 1586-1594, Nov.

[7] D.R. Brownbridge, L.F. Marshall, and B. Randell, "The Newcastle Connection or Unixes of the World Unite!," Software Practice and Experience, vol. 12, no. 12, pp. 1147-1162, 1982..

[8] R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, and B. Lyon, "Design and Implementation of the Sun Network Filesystem," Proc. USENIX Assoc. Conf., 1985

[9] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable Secure File Sharing on Untrusted Storage," Proc. Second USENIX Conf. File and Storage Technologies (FAST), pp. 29- 42, 2003.

[10] S.C. Rhea, P.R. Eaton, D. Geels, H. Weatherspoon, B.Y. Zhao, and J. Kubiatowicz, "Pond: The Oceanstore Prototype," Proc. Second USENIX Conf. File and Storage Technologies (FAST), pp. 1-14, 2003