International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 8, August 2020||

Efficient Public Verification Data Integrity Checking On Cloud

Dr Shyam Gupta[1], Pramila Shahaji Gite[2]

Associate Professor, Department of Computer Engineering, Siddhant College of Engineering MH, India[1]

PG Student, Department of Computer Engineering, Siddhant College of Engineering MH, India[2]

ABSTRACT:Cloud storage services is allow users to store their data on cloud servers to the save local data storage costs. However, unlike using a native storage devices, users are don't physically manage the data stored on cloud servers; so the information integrity of the outsourced data has been become a problem. Several public auditing verification schemes is proposed to enable a third-party auditor to verify the data integrity for the users. These schemes create an impractical assumption. The auditors have been enough computation capability in the reality expensive verification costs. In this paper, we planned a unique public verification scheme for the cloud storage using in distinguish with the ability obfuscation, which requires a light-weight computation on the auditor and delegate most of computation to the cloud. With the development of cloud storage, how to protect the integrity of the data stored in the cloud becomes the clients' major concern. Recently, Zhang et al. proposed a certificate less data integrity auditing scheme for cloud-assisted cyber-physical-social systems. However, we discover that their theme has some security flaws, i.e., if a client outsources a file with m blocks data and tags, the cloud server will store only one block of them, and pass the integrity auditing of the auditor. We refer to this attack as storing one block attack. Then, we offer an improved scheme to amend these flaws.

KEYWORDS: Certificate-less public key cryptography, cloud storage, data integrity auditing, provable data possession (PDP), storing one block attack.

I. INTRODUCTION

With the development of the cloud storage, more and more people intend to source their data into cloud server. However, without the dominant of their data, the cloud service provider could delete some client's data for saving memory space or mask their information corruption event for his/her name. Therefore, how to protect the integrity of the data stored in the cloud becomes the clients' major concern. Some information verification schemes think about users themselves to execute the verification. However, this implies that a user needs to expend further communication and computation costs to verify information integrity. As a consequence of this, the user is needed in-tuned heavy communication and verification burden to retrieve and use the information. To reduce the verification burden on users, a public verification paradigm has been planned. The concept is to use associate external and freelance auditor to sporadically verify the information integrity on behalf of users. As a result, users are free from the verification burden whereas the auditor must be equipped with robust computation capability for verification. In existing public verification schemes, the computation overhead of verification by the auditor linearly will increase with the scale of the verified information set. If the verification is needed to be dead often for multiple users' information sets, the auditor can want a large computation capability to accomplish the verifications and also the verification delay are huge. The readying of such auditor is so a tough drawback. Delay on the auditor aspect contains a vital value to make the verification scheme efficient and practical. We construct a unique cloud storage auditing scheme for shared information supporting real economical user revocation in this paper.

So as to understand economical user revocation, we come up with a unique strategy for key generation. In this design, the group's public key is replaced by the group's identity data, that remains unchanged within the whole One part remains mounted since being issued, and the other part alters with user revocation. We also propose a unique non-public key update technique to support user revocation. Once users are revoked from the cluster, all of the non-revoked users will update their non-public keys by this technique to form the cloud storage auditing still work, while the identity data of the cluster doesn't have to amendment. In addition, the revoked users aren't ready to transfer information and authenticators to the cloud any longer. During this approach, all of the authenticators generated before user revocation don't have to be recomputed. Therefore, the overhead of user revocation is fully freelance of the whole range of the revoked user's blocks.

Even once the number of knowledge is brobdingnagian, the group can still complete user revocation terribly expeditiously. Besides, our theme is predicated on identity-based cryptography, which eliminates the difficult

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 8, August 2020||

certificate management in ancient PKI systems, as well as certificate generation, certificate revocation, certificate renewal, etc.

II. RELATED WORK

A. Provable data possession at untrusted stores:-In this paper introduce a model for obvious knowledge possession (PDP) that enables a consumer that has hold on knowledge at an untrusted server to verify that the server possesses the initial knowledge while not retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server that drastically reduces I/O costs. The consumer maintains a constant quantity of data to verify the proof. The challenge/response protocol transmits a small, constant quantity of knowledge that minimizes network communication. Thus, the PDP model for remote knowledge checking supports large knowledge sets in widely-distributed storage systems. They tend to present 2 provably-secure PDP schemes that are additional economical than previous solutions, even compared with schemes that succeed weaker guarantees. Above all, the overhead at the server is low (or even constant), as against linear within the size of the data. Experiments using our implementation verify the usefulness of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation. [1]

B. Proxy provable data possession in public clouds:-Recently, cloud computing rapidly expands as another to standard computing because of it will offer a flexible, dynamic and resilient infrastructure for each educational and business environments. In public cloud environment, the consumer moves its information to public cloud server (PCS) and might not management its remote information. Thus, data security is an important problem publicly cloud storage, like information confidentiality, integrity, and availability. In some cases, the consumer has no ability to examine its remote information possession, like the consumer is in prison thanks to committing crime, on the ocean-going vessel, within the battle ground because of the war, et al. it's to delegate the remote information possession checking task to some proxy. During this paper, they study proxy provable information possession (PPDP). Publicly clouds, PPDP could be a matter of crucial importance once the consumer can't perform the remote information possession checking. They tend to study the PPDP system model, security model and style methodology. Supported the bilinear pairing technique, they tend to design an economical PPDP protocol. Through security analysis and performance analysis, our protocol is demonstrable secure and economical.[2]

C. Identity-based distributed provable data possession in multicloud storage:-Remote information integrity checking is of crucial importance in cloud storage. It will build the clients verify whether or not their outsourced information is unbroken intact while not downloading the entire information. In some application situations, the clients got to store their information on multi-cloud servers. At an equivalent time, the integrity checking protocol should be economical so as to save lots of the verifier's cost. From the 2 points, they have a tendency to propose a novel remote information integrity checking model: ID-DPDP (identity-based distributed demonstrable information possession) in multi-cloud storage. The formal system model and security model area unit given. Supported the bilinear pairings, a concrete ID-DPDP protocol is designed. The planned ID-DPDP protocol is demonstrably secure under the hardness assumption of the quality CDH (computational Diffie Hellman) drawback. Additionally to the structural advantage of elimination of certificate management, our ID-DPDP protocol is also economical and versatile. supported the client's authorization, the projected ID-DPDP protocol will understand non-public verification, delegated verification and public verification.[3]

D. Candidate in distinguishability obfuscation and functional encryption for all circuits:-In this paper, introduce a in distinguishability obfuscation and functional encryption for general circuits: In distinguishability obfuscation requires that given any two equivalent circuits C0 and C1 of similar size, the obfuscations of C0 and C1 should be computationally indistinguishable. In functional encryption, cipher texts encrypt inputs x and keys are issued for circuits C. Using the key SKC to decrypt a cipher text CTx = Enc(x), yields the value C(x) but does not reveal anything else about x. Furthermore, no collusion of secret key holders should be able to learn anything more than the union of what they can each learn individually. They give constructions for in distinguishability obfuscation and functional encryption that supports all polynomial-size circuits. They accomplish this goal in three steps: - • they describe a candidate construction for in distinguishability obfuscation for NC1 circuits. The security of this construction is based on a new algebraic hardness assumption. The candidate and assumption use a simplified variant of multilinear maps, which they call Multilinear Jigsaw Puzzles. • They show how to use in distinguishability obfuscation for NC1 together with Fully Homomorphic Encryption (with decryption in NC1) to achieve in distinguishability obfuscation for all circuits. • Finally, they show how to use in distinguishability

obfuscation for circuits, public-key encryption, and non-interactive zero knowledge to achieve functional encryption for all circuits. The functional encryption scheme they construct also enjoys succinct cipher texts, which enables several other applications.[11]

E. Certificate less public auditing for data integrity in the cloud:-Due to the existence of security threats within the cloud, several mechanisms are planned to allow a user to audit information integrity with the general public key of {the information|the info|the information} owner before utilizing cloud data. The correctness of selecting the right public key in previous mechanisms depends on the protection of Public Key Infrastructure (PKI) and certificates. Though traditional PKI has been wide employed in the development of public key cryptography, it still faces several security risks, particularly within the aspect of managing certificates. During this paper, they have a tendency to style a certificate-less public auditing mechanism to eliminate the protection risks introduced by PKI in previous solutions. Specifically, with our mechanism, a public verifier does not have to manage certificates to decide on the correct public key for the auditing. Instead, the auditing will be operated with the help of the info owner's identity, like her name or email address, which might ensure the right public key is used. Meanwhile, this public supporter continues to be ready to audit information integrity while not retrieving the complete information from the cloud as previous solutions. To the most effective of our data, it's the primary certificate-less public auditing mechanism for validating information integrity within the cloud. Our theoretical analyses prove that our mechanism is correct and secure, and our experimental results show that our mechanism is in a position to audit the integrity of information within the cloud efficiently.[5]

F. Short signatures from the wail pairing:-In this paper introduce a short signature scheme based on the Computational Diffie– Hellman assumption on certain elliptic and hyper elliptic curves. For standard security parameters, the signature length is about half that of a DSA signature with a similar level of security. Our short signature scheme is designed for systems where signatures are typed in by a human or are sent over a low-bandwidth channel. They survey a number of properties of our signature scheme such as signature aggregation and batch verification[13]

G. Scalable and efficient provable data possession:- Storage outsourcing could be a rising trend that prompts variety of interesting security problems, several of that are extensively investigated within the past. However, provable information Possession (PDP) could be a topic that has only recently appeared within the analysis literature. The most issue is the way to often, with efficiency and securely verify that a storage server is reliably storing its client's (potentially very large) outsourced information. The storage server is assumed to be untrusted in terms of each security and responsibility. (In alternative words, it would maliciously or accidentally erase hosted data; it would conjointly relegate it to slow or off-line storage.) The matter is exacerbated by the consumer being a small electronic computer with restricted resources. This drawback using either public key cryptography or requiring the consumer to the source its information in encrypted kind. During this paper, they tend to construct a extremely economical and provable secure PDP technique based mostly entirely on symmetric key cryptography, whereas not requiring any bulk coding. Also, in distinction with its predecessors, our PDP technique allows outsourcing of dynamic information, i.e., it with efficiency supports operations, like block modification, deletion and append. [15]Identity-Based Proxy-Oriented Data Uploading and Remote Data Integrity Checking in Public Cloud:-More and additional clients would really like to store their information to PCS (public cloud servers) beside the fast development of cloud computing. New security issues have to be resolved so as to help additional clients method their information publicly cloud. Once the client is restricted to access PCS, he can delegate its proxy to method his information and upload them. On the opposite hand, remote information integrity checking is additionally a very important security drawback publicly cloud storage. It makes the clients check whether their outsourced information is kept intact while not downloading the total information. From the protection issues, they propose a completely unique proxy-oriented information uploading and remote information integrity checking model in identity-based public key cryptography: IDPUIC (identity-based proxy-oriented information uploading and remote information integrity checking publicly cloud).They provide the formal definition, system model and security model. Then, a concrete ID-PUIC protocol is designed by using the bilinear pairings. The proposed ID-PUIC protocol is demonstrably supported the hardness of CDH (computational Diffie-Hellman) problem. Our ID-PUIC protocol is additionally efficient and flexible. Supported the first client's authorization, the projected ID-PUIC protocol will understand personal remote information integrity checking, delegated remote information integrity checking and public remote information integrity checking. [4]

H. Certificate less Public Auditing Protocol with Constant Verification Time:-To provide the integrity of outsourced information within the cloud storage services, several public auditing schemes which permit a user to examine the integrity of the outsourced information are planned. Since most of the schemes are constructed on Public Key Infrastructure (PKI), they suffer from many concerns like management of certificates. To resolve the issues,

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 8, August 2020||

certificate less public auditing schemes even have been studied in recent years. During this paper, they propose a certificate less public auditing theme that has the constant-time verification algorithm. Therefore, our scheme is more economical than previous certificate less public auditing schemes. To prove the security of our certificate less public auditing theme, they first outline three formal security models and prove the security of our theme under the three security models. [6] Multiparty Key Exchange, Efficient Traitor Tracing, and More from in distinguishability Obfuscation:-they show how to use in distinguishability obfuscation to build multiparty key exchange, efficient broadcast encryption, and efficient traitor tracing. Our schemes enjoy several interesting properties that have not been achievable before: • our multiparty non-interactive key exchange protocol does not require a trusted setup. Moreover, the size of the published value from each user is independent of the total number of users. • Our broadcast encryption schemes support distributed setup, where users choose their own secret keys rather than be given secret keys by a trusted entity. The broadcast cipher text size is independent of the number of users. • Our traitor tracing system is fully collusion resistant with short cipher texts, secret keys, and public key. Cipher text size is logarithmic in the number of users and secret key size is independent of the number of users [18].

III. METHODOLOGY

We use (1) Cloud: The cloud provides enormous storage space and computing resources for group users. Through the cloud storage, group users can enjoy the data sharing service. (2)TPA: The TPA is responsible for auditing the integrity of cloud data on behalf of group users. When the TPA wants to audit the data integrity, it will send an auditing challenge to the cloud. After receiving the auditing challenge, the cloud will respond to the TPA with a proof of data possession. Finally, the TPA will verify the data integrity by checking the correctness of the proof. The TPA is a powerful party and it is honest. (3)Cloud Users. The users are the data owners, who outsource their data to the cloud server and need to access the data flexibly. The integrity of the cloud-stored data should be verified periodically. However, due to limited condition, the users cannot check the data integrity by themselves. Therefore, the users will resort to a third-party auditor (TPA) to perform the verification work

A. Architecture

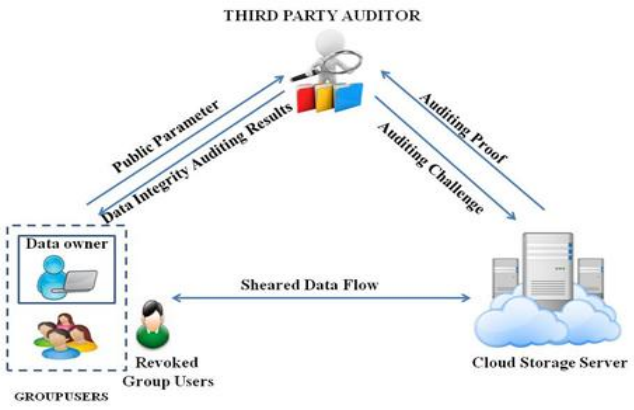

Fig: Proposed System Architecture

In our system model, the shared information belong to the dynamic group composed of non-revoked users. Everybody throughout this dynamic group can transfer information and share them with different group users. Once a user is revoked, these information uploaded by it are still shared by the dynamic group. The owner of those information still are this group. However, the revoked user wouldn't be ready to transfer information and thus the corresponding authenticators to the cloud any additional. User send file to store on cloud.TPA generate 160 bit of tag using SHA-1 algorithmic rule. And encrypted using AES algorithm so send to cloud server for store the file. When user wants file from cloud, it send request to TPA.TPA check the tag that receives from cloud and send to user.

Algorithms:

(1) SHA-1 (Secure Hash Algorithm) is a cryptographic hash function which takes an input and produces a 160-bit (20-byte) hash value known as a message digest – typically rendered as a hexadecimal number, 40 digits long. This is designed to be computationally infeasible to:

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 8, August 2020||

a) Obtain the original message, given its message digest.
b) Find two messages producing the same message digest.
Each round takes 3 inputs-
• 512-bit block,
• The register abcde
• A constant K[t] (where t= 0 to 79)

SHA-1 Pseudo-code:

1).The input message is broken up into chunks of 512-bit blocks (sixteen 32-bit little endian integers), the message is padded so that its length is divisible by 512.
2).The padding works as follows: first a single bit, 1, is appended to the end of the message.
3).This is followed by as many zeros as are required to bring the length of the message up to 64 bits less than a multiple of 512.
4).The remaining bits are filled up with a 64-bit integer representing the length of the original message, in bits.
5).The MD5 algorithm uses 4 state variables, each of which is a 32 bit integer (an unsigned long on most systems). These variables are sliced and diced and are (eventually) the message digest.
The variables are initialized as follows:
A = 0x67452301
B = 0xEFCDAB89
C = 0x98BADCFE
D = 0x10325476.
6). Now on to the actual meat of the algorithm: the main part of the algorithm uses four functions to thoroughly goober the above state variables. Those functions are as follows:
F(X, Y, Z) = (X & Y) | (~(X) & Z)
G(X, Y, Z) = (X & Z) | (Y & ~(Z))
H(X, Y, Z) = X ^ Y ^ Z
I(X, Y, Z) = Y ^ (X | ~(Z))
Where &, |, ^, and ~ are the bit-wise AND, OR, XOR, and NOT operators
7). These functions, using the state variables and the message as input, are used to transform the state variables from their initial state into what will become the message digest. For each 512 bits of the message, the rounds performed (this is only pseudo-code, don't try to compile it)
After this step, the message digest is stored in the state variables (A, B, C, and D). To get it into the hexadecimal form you are used to seeing, output the hex values of each the state variables, least significant byte first. For example, if after the digest:
A = 0x01234567;
B = 0x89ABCDEF;
C = 0x1337D00D
D = 0xA5510101
Then the message digest would be:
67452301EFCDAB890DD03713010151A5 (required hash value of the input value).

(2) AES (Advanced Encryption Standard)
The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES).
1. Derive the set of round keys from the cipher key
2. Initialize the state array with the block data (plaintext).
3. Add the initial round key to the starting state array.
4. Perform nine rounds of state manipulation.
5. Perform the tenth and final round of state manipulation
6. Copy the final state array out as the encrypted data

AES Pseudo-code:

```
Cipher (byte in[16], byte out[16], key_array round key[Nr+1])
begin
{
byte state[16];
```

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 8, August 2020||

```
state = in;
AddRoundKey(state, round_key[0]);
for( i = 1 to Nr-1 stepsize 1) do
{
SubBytes(state);
ShiftRows(state);
MixColumns(state);
AddRoundKey(state, round_key[i]);
end for
SubBytes(state);
ShiftRows(state);
AddRoundKey(state, round_key[Nr]);
end
}
}
```

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

Compared with previous public verification schemes, our scheme requires the cloud server to spend more computation resources for data integrity verification. Since the cloud server provides an on-demand, flexible, and usage-based pay-for-use service for users, and our scheme can be extended to other POR/PDP-based public verification scheme without reprocessing the cloud-stored data, our scheme not only enables the cloud server to provide a new option for users with well compatibility, but also brings additional financial benefits for cloud server (service provider). Therefore, the cloud service provider could be motivated to collaborate with users to reduce the verification overhead on the auditor side.

Fig: Result 1

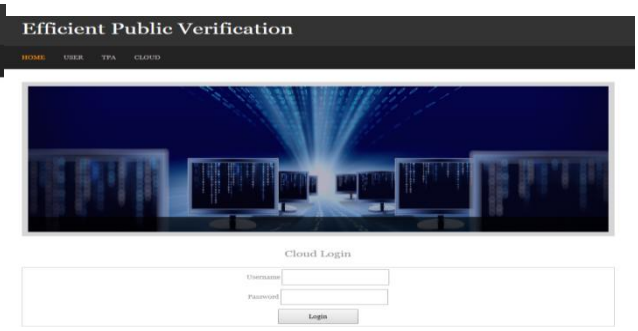

Fig: Result 2

Fig: Result 3

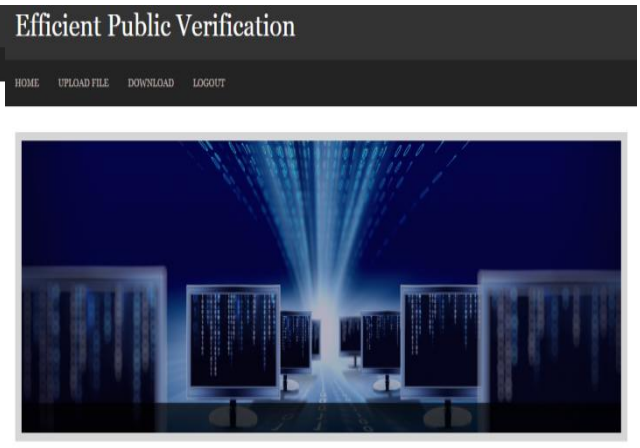

Fig: Result 4

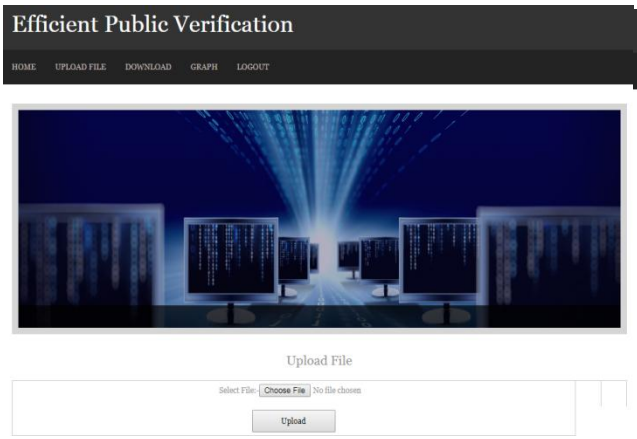

Fig: Result 5

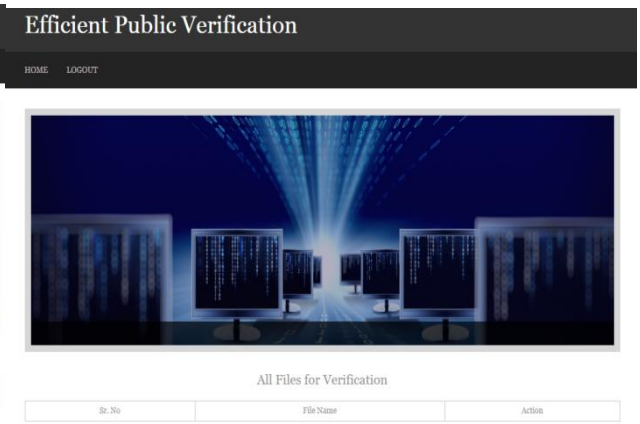

Fig: Result 6

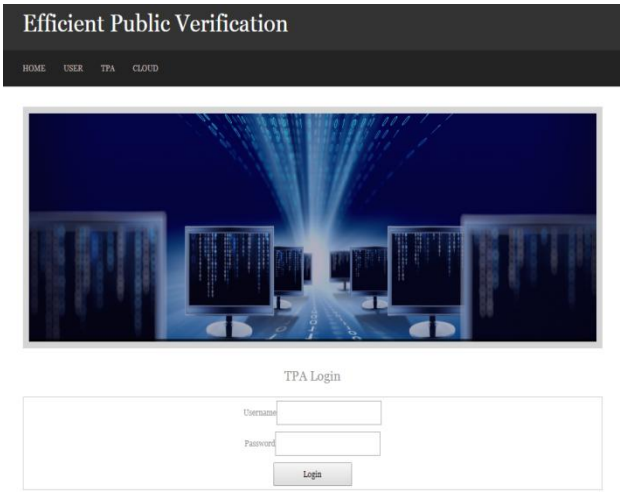

Fig: Result 7

V. CONCLUSION

This system is implemented to achieve the verification in efficient public integrity auditing on the data. The advantages of the protocol are store secure identical files on cloud .Also encrypted information is achieved and are protected from internal and external attacks. We propose a system for security purpose. TPA should efficiently verify the distributed information storage without requesting for the local copy of information. Cloud server is used for store the tag of files in encrypted form.

VI. CONTRIBUTION OF WORK

In contribution we use proxy server for regenerate the file. Proxy server is an intermediary server between client and the internet. Since the proxy server hides the identity of the user hence it protects from spam and the hacker attacks. In this system proxy server is used for security purpose. Cloud receives files from the user. If hacker hack the file then this file send to proxy server for regenerate the file. Thus the main function of proxy server is regenerate files. We also contribute the concept of identical files on cloud, is a compression technique that aims to remove duplicate information from a dataset. Data duplication frees up a lot of storage particularly when it is performed over large volumes of data. The data duplication process works using software that eliminates redundant data and storing only unique first instances of any data. Any other copies of the same data found by the software are removed, and in their place will be a pointer or reference to the original copy.

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

||Volume 9, Issue 8, August 2020||

REFERENCES

[1] G. Ateniese et al., "Provable data possession at untrusted stores," in Proc. CCS, 2007, pp. 598–609.
[2] [2] H. Wang, "Proxy provable data possession in public clouds," IEEE Trans. Services Comput., vol. 6, no. 4, pp. 551–559, Oct./Dec. 2013.
[3] [3] H. Wang, "Identity-based distributed provable data possession in multicloud storage," IEEE Trans. Services Comput., vol. 8, no. 2, pp. 328–340, Mar./Apr. 2015
[4] H. Wang, D. He, and S. Tang, "Identity-based proxy-oriented data uploading and remote data integrity checking in public cloud," IEEE Trans. Inf. Forensics Secur., vol. 11, no. 6, pp. 1165–1176, Jun. 2016.
[5] B. Wang, B. Li, H. Li, and F. Li, "Certificateless public auditing for data integrity in the cloud," in Proc. CNS, Oct. 2013, pp. 136–144.
[6] D. Kim and I. R. Jeong, "Certificateless public auditing protocol with constant verification time," Secur. Commun. Netw., vol. 2017, Jan. 2017, Art. no. 6758618, doi: 10.1155/2017/5758618.
[7] Y. Zhang, C. Xu, S. Yu, H. Li, and X. Zhang, "SCLPV: Secure certificateless public verification for cloud-based cyber-physical-social systems against malicious auditors," IEEE Trans. Comput. Social Syst., vol. 2, no. 4, pp. 159–170, Dec. 2015.
[8] D. Cash, A. Kupc¸ ¨ u, and D. Wichs, "Dynamic proofs of retrievability ¨ via oblivious ram," in Proceedings of EUROCRYPT. Springer, 2013, pp. 279–295.
[9] [9]E. Shi, E. Stefanov, and C. Papamanthou, "Practical dynamic proofs of retrievability," in Proceedings of CCS. ACM, 2013, pp. 325–336
[10] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in Proceedings of INFOCOM. IEEE, 2010, pp. 1–9
[11] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters, "Candidate indistinguishability obfuscation and functional encryption for all circuits," in Proceedings of FOCS. IEEE, 2013, pp. 40–49.
[12] B. Wang, B. Li, and H. Li, "Panda: Public auditing for shared data with efficient user revocation in the cloud," IEEE Transactions on Services Computing, vol. 8, no. 1, pp. 92–106, 2015.
[13] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in Proceedings of ASIACRYPT. Springer, 2001, pp. 514–532.
[14] Y. Zhang, C. Xu, S. Yu, H. Li, and X. Zhang, "Sclpv: Secure certificateless public verification for cloud-based cyber-physical-social systems against malicious auditors," IEEE Transactions on Computational Social Systems, vol. 2, no. 4, pp. 159–170, 2015.
[15] [15]G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proceedings of SecureComm. ACM, 2008
[16] [16] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 9, pp. 1717–1726, 2013.
[17] [17] M. Sookhak, A. Gani, M. K. Khan, and R. Buyya, "Dynamic remote data auditing for securing big data storage in cloud computing," to appear, doi: 10.1016/j.ins.2015.09.004.
[18] [18] D. Boneh and M. Zhandry, "Multiparty key exchange, efficient traitor tracing, and more from indistinguishability obfuscation," in Proceedings of CRYPTO. Springer, 2014, pp. 480–499
[19] [19] C. Erway, A. Kupc¸ ¨ u, C. Papamanthou, and R. Tamassia, "Dynamic ¨ provable data possession," ACM Transactions on Information and System Security, vol. 17, no. 4, p. 15, 2015.