ISSN:2348-2079

Volume-6 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

On The Security Of Data Access Control For Multi Authority Cloud Storage Systems

[1] Nivodhini M K, [2]Vasuk.P, [3]Kiruthika K, [4]Lokesh P N, [5]Sangeetha C, [6]Suriya Kumar S

[1]Assistant Professor, Department of Computer Science and Engineering, K.S.R. College of Engineering,

[2]Assistant Professor, Department of Computer Science and Engineering, K.S.R. College of Engineering,

[3-6]UG Student, Department of Computer Science and Engineering, K.S.R. College of Engineering.

E-mail id: [1]nivodhinimk99@gmail.com, [2]vasukiabi@gmail.com , [3]kiruthika.k1413048@ksrce.ac.in, [4]lokeshwtf96@gmail.com, [5]sangeethaaviji@gmail.com, [6]suriyakumar.s19@gmail.com,

***ABSTRACT* – Data Access control has becoming a challenging issue in cloud storage systems. Some technique have been proposed to achieve the secure data access control in semi trusted cloud storage system. Recently, K. Yang et al. proposed a basic access control scheme for multi authority cloud storage system (DAC-MAC) and an extensive data access control scheme (EDAC-MACS). We claimed that the DAC-MACS cloud achieve efficient decryption and immediate revocation and the EDAC-MACS could achieve these goals even though no revoked users reveals their key Update keys to the revoked user. However, though our cryptanalysis, the revocation security of both schemes cannot be guaranteed.**

Keywords – Data Access control, Multi-authority cloud storage, Efficient Decryption, Revoction Security.

I. Introduction

Secure data management in cloud computing providing an efficient multi-receiver IBE scheme that only requires "one" (or "none") pairing computation to encrypt a single message for multiple receivers. We provide formal security notions for multi-receiver IBE schemes based on the "selective identity attack" model in which an attacker outputs ahead of time the identities of multiple receivers that it wishes to challenge. We then prove that our schemes are secure against chosen plaintext attack (CPA) and adaptive cipher text attack ("ACTA") in the random oracle model assuming the standard assumptions related to the Bilinear Diffie-Hellman problems are computationally hard. Finally, we show how our schemes lead to very efficient public key broadcast encryption schemes based on the "subset-cover" framework. As an independent interest, We discuss how the selective identity attack model plays an important role in obtaining an efficient reduction in the security analysis of our efficient multi-receiver IBE schemes. Data Extraction is an analytic process designed to explore data (usually large amounts of data-typically business or market related) in search of consistent patterns and/or systematic relationships between variables, and then to validate the findings by applying the detected patterns to new subsets of data. The ultimate goal of data mining is prediction - and predictive data mining is the most common type of data mining and one that has the most direct business applications. The process of data mining consists of three stages: (1) the initial exploration, (2) model building or pattern identification with validation/verification, and (3) deployment (i.e., the application of the model to new data in order to generate predictions.

II. RELATED WORK

As mentioned above, CP-ABE is a promising cryptographic mechanism for fine-grained access control. Bethen court et al. explicitly formalized the notion of CP-ABE and proposed a CP-ABE scheme but its security proof was given in the generic group model. Cheung and Newport proposed another CP-ABE scheme that supports AND ∗, +, − access policy, and proved its security under decision bilinear Diffie Hellman assumption. Later, a number of CP-ABE schemes were proposed for better efficiency, or security, or expressiveness. The first multi-authority ABE (MA-ABE) scheme was proposed by Chase. there are several AAs and one central authority (CA) in the system. Each AA issues a set of attribute secret keys to each user, while the CA distributes a global unique identifier together with a final secret key to each user. Other multi-authority ABE schemes have been proposed. Emura et al. put forth a CP-ABE scheme with constant-size cipher text. And yet, their scheme only supports the (n, n)-threshold access policy on multi-valued attributes. Another CP-ABE scheme with constant-size cipher text was proposed, and works for the (t, n)-threshold case proposed two new CP-ABE schemes, which have both constant-size cipher text and small computation cost for AND∗ +, − access policy. Sreenivasa and Dutta proposed the first fully security CP-ABE scheme with constant-size cipher text by adopting the technique of over composite order bilinear group. The revocation issue is an important and cumbersome problem in

attribute-based systems. Several CP-ABE schemes which support attribute-level revocation have been proposed. For attribute-level revocation, any revoked user only loses part access privileges as some attributes are removed. That is, each revoked user can still access the data as long as his/her remaining attributes satisfy the access policy. Besides binding an expiration time to each attribute, the revocation methods in CP-ABE schemes can be classified into two categories: directly revocation and indirectly revocation In the direct revocation, the AA publishes the revocation list so that users can integrate revocation information into the cipher text while encrypting data. A non-revoked user can decrypt the cipher text only if the attributes of that user satisfy the access policy in the cipher text. The advantage of this method is that the attribute-level revocation can be enabled without updating attribute secret keys for the non-revoked users. In the indirect revocation, the AA needs to update the secret key with respect to the revoked attribute for each non-revoked user, instead of making the revocation list public to users. Concretely, Zhang et al. drew support from an auxiliary function to indicate which cipher texts are involved in revocation events to update these involved cipher texts. Yu et al. proposed a CP-ABE scheme with indirect attribute-level revocation by the semi-trusted proxy deployed in the data server. The key re-randomization is adopted in Yang et al.'s CP-ABE scheme Hur and Noh proposed an immediate attribute-level revocation mechanism in CP-ABE by utilizing a binary key-encrypted-key tree for attribute group key distribution. Different from the attribute-level revocation, user level revocation makes the revoked users lose all the access privileges in the system. In, Attrapadung and Imai proposed a CP-ABE scheme with direct user-level revocation by combining the techniques of broadcast encryption and ABE.

III. PROPOSED SYSTEM

The advantages over the existing system are, we use an identity tree instead of key tree in our scheme. Each node in the identity tree is associated with an identity. The leaf node's identity is corresponding to the user's identity and the intermediate node's identity is generated by its children's identity. Hence, in an identity tree, an intermediate node represents set users in the sub tree rooted at this node. We propose a novel multi-cloud Authentication protocol, namely IBE, including two schemes. The basic scheme (IBE) eliminates the correlation among data's and thus provides the perfect resilience to data security, and it is also efficient in terms of latency, computation, and communication overhead due to an efficient cryptographic primitive called batch signature, which supports the authentication of any number of Data simultaneously. We also present an enhanced scheme IBE-E, which combines the basic scheme with a data filtering mechanism to the DoS impact while preserving the perfect resilience to data security. The keys used in each subgroup can be generated by a group of IBE on Multi cloud storage Key Generation centers (IBE) in parallel. All the members in the same subgroup can compute the same subgroup key though the keys for them are generated by different KGCs. This is a desirable feature especially for the large-scale network systems, because it minimizes the problem of concentrating the workload on a single entity. Data Format Independence PDP schemes put no restriction on the format of the data in particular files stored at the server do not have to be encrypted. This feature is very relevant since we anticipate that PDP schemes will have the biggest impact when used with large public repositories (e.g., digital libraries, astronomy/medical/legal repositories, archives etc.).Prime-order Group Variant PDP schemes can potentially be modified to work within a group of a publicly-known prime order q. In this case, however, file blocks (seen as integers) must be less than q, otherwise the server could simply store them reduced modulo q. In a prime-order setting, network communication is further reduced (particularly in the elliptic curve setting), but pre-processing becomes more expensive given the small size of the file blocks. In contrast, the RSA setting allows us to work with arbitrarily large file blocks. Multiple Files have described PDP schemes for the case when a client stores a single file F on the server. In the Tag Block algorithm, for each block mi the client computes a tag over the tuple (Wi ,mi). We emphasize that the values Wi cannot be reused. Since Wi is obtained by concatenating a secret value v with i, this implies that the indices i must be different across all tags. In other words, the client should not use the same index twice for computing tags. This condition holds because in our scheme an index i is simply the position of the block mi in the file. In order to store multiple files on the server, the client must ensure that indices used to compute tags are distinct not only across the tags corresponding to the blocks of each file, but also across the tags corresponding to the blocks of all files. One simple method to achieve this is to prepend the file's identifier to the index. For example, if the identifier of a file F = (m1,... ,mn) is given by id(F), then for each block mi , $1 \leq i \leq n$, C computes the tag (Tid(F)||i,mi , Wid(F)||i) ← Tag Block(pk,sk,mi , id(F)||i). The uniqueness of indices is ensured under the assumption that each file has a unique identifier. Another simple way to ensure that indices are only used once is to use a global counter for the index, which is incremented by the client each time after a tag is computed. Protocol steps perform the same computation. It generates data at about 433 KB/s on average. The preprocessing performance of B-PDP differs from the challenge phase even though both steps compute the exact same signature. This is because the client has access to φ(N) and can reduce the file modulo φ(N) before exponentiation. In contrast, the security of the protocol depends on φ(N) being a secret that is unavailable to the server. The preprocessing costs comprise a single exponentiation and computing a modulus against the entire file. E-PDP also exponentiation data that was reduced modulo φ(N) but does not reap the same speed up, because it must do so for every block. This creates a natural trade-off between preprocessing time and challenge time by varying the block size; e.g., the protocol devolves to B-PDP for files of a single block. We choose a block size of 4K in order to minimize the server's effort. Given the efficiency of computing challenges, pre-processing represents the limiting performance factor for E-PDP. The rate at which clients can generate data to outsource bounds the overall system performance perceived by the client. However, there are several mitigating factors. Outsourcing data is a one-time task, as compared to challenging outsourced data, which will be done repeatedly. The process is completely parallelizable. Each file can be

processed independently at a different processor. A single file can be parallelized trivially if processors share key material.

A. KEY AGREEMENT IN PEER GROUPS

A number of group key management techniques have been proposed in the past. We generally fall into three categories: (1) centralized, (2) distributed, and (3) contributory.
Centralized group key management is conceptually simple as it involves a single entity (or a small set of entities) that generates and distributes keys to group members via a pair-wise secure channel established with each group member. We view centralized group key management as inappropriate for secure peer group communication, since a central key server must be, at the same time, continuously available and present in every possible subset of a group in order to support continued operation in the event of arbitrary network partitions. Continuous availability can be addressed by using fault-tolerance and replication techniques. Unfortunately, the present issue is difficult to solve in a scalable and efficient manner.

B. DISTRIBUTED GROUP KEY MANAGEMENT

It is more suitable to peer group communication, especially over unreliable networks. It involves dynamically selecting a group member that acts as a key distribution server. Although robust, this approach has a notable drawback in that it requires a key server to maintain long-term pair-wise secure channels with all current group members in order to distribute group keys. Some schemes take advantage of data structures to minimize the number of encryption and messages that must be generated when the key changes. When a new key server is selected all these data structures also need to be recreated.

C. CONTRIBUTORY GROUP KEY AGREEMENT

It requires every group member to contribute an equal share to the common group secret, computed as a function of all members' contributions. These protocols are appropriate for dynamic peer groups. This approach avoids the problems with the single point(s) of trust and failure. Moreover, some contributory methods do not require establishing pair-wise secret channels among group members. Also, unlike most group key distribution protocols, We offer strong key management security properties such as key independence and perfect forward secrecy (PFS). Recent research on authenticated group key agreement protocols provides stronger security guarantees against active attackers. More detailed discussion can be found in. We note that many centralized and distributed key management protocols (such as the Logical Key Hierarchy (LKH) Protocol, One-Way Function Tree protocol and Centralized Flat Table, to rely on symmetric encryption to distribute group keys, as opposed to contributory protocols which rely on modular exponentiations. Therefore, We do not provide PFS. However, such protocols scale to large groups and have a lighter overhead than contributory ones. The cost of group key management is determined by two dominating factors: communication and computation. Typically, efficiency in one comes at the expense of the other. Protocols that distribute computation usually require more communication rounds, while protocols minimizing communication require more computational effort.

D. GROUP KEY MANAGEMENT

As noted above, the focus of this work is on the performance of group key management protocols for collaborative peer groups. Therefore, we consider only distributed key distribution and contributory key agreement protocols. In looking at the available protocols, we are concerned mostly with the cost (performance) of the types of group key management operations that occur most often. At the first glance, it might appear that a typical collaborative group scenario is as follows: a group forms, functions for some time, and then dissolves itself. If this were true, we would only need to consider the performance of the initial key agreement leading to the group's formation. Moreover, performance would not be of great concern because the protocol would be invoked only once or very infrequently in order to rekey the group. However, a typical collaborative group is formed incrementally and its population can mutate throughout its lifetime due either to members joining and leaving or to network connectivity changes. It begins with the STR protocol proposed by and originally aimed at teleconferencing. As will be seen later, STR is well suited for adding new members as it takes only two rounds and two modular exponentiations. However, member exclusion (rekeying following a member leave event) is relatively inefficient. In proposed an efficient protocol which takes only two rounds and three modular exponentiations per member to generate a group key. This protocol allows all members to recompute the group key for any membership change with a constant CPU cost. The distribution in computation is obtained at the cost of using $2n$ broadcast messages which is expensive on a wide-area network. In proposed an authenticated key agreement scheme based on secure multiparty computation. This protocol uses two communication rounds, but each round consists of n simultaneous broadcast messages. Although the cryptographic mechanisms are quite elegant, the main shortcoming is the lack of PFS. Steiner et al. addressed dynamic membership issues in group key agreement as part of developing a family of GDH protocols based on straightforward extensions of the two-party Diffie–Hellman protocol. GDH protocols are relatively efficient for member leave and group partition operations, but the merge protocol requires the number of rounds equal to the number of new (merging) members. Yielded a TGDH protocol, which is more efficient than GDH in both communication and computation. It has performed as in-depth performance analysis on CKD protocols. The authors show that refreshing group key periodically by aggregating membership events provides better efficiency than refreshing group key for every membership event. In general, this can be true for peer group scenario. However, the former provide weaker security than the latter. It focus is on group key management over peer group, while their focus is on centralized key management.

E. LITERATURE REVIEW

Y. Amir et al has proposed Group key agreement is a fundamental building block for secure peer group communication systems. Several group key management techniques were predictable in the last decade, all assuming the survival of an underlying group communication infrastructure to provide dependable and ordered message delivery as well as group membership information. Despite analysis, implementation, and utilization of some of these techniques, the actual costs allied with group key management have been poorly understood so far. This resulted in an uninvited tendency: on the one hand, adopting suboptimal precautions for consistent group communication, while, on the other hand, constructing excessively precious group key management protocols. This work presents a thorough performance appraisal of five notable circulated key management techniques (for collaborative peer groups) built-in with a reliable group communication system. An in-depth comparison and assessment of the five techniques is obtainable based on investigational results obtained in actual local- and wide-area networks. The extensive concert measurement experiments conducted for all methods offer insights into their scalability and practicality. Furthermore, their analysis of the experimental results places of interest several observations that are not evident from the theoretical analysis. D.Augot et al has projected a Group Key Agreement (GKA) etiquette is a mechanism to launch a cryptographic key for a group of participants, based on each one's contribution, over a public network. The key, thus derived, can be used to establish a secure channel between the participants. When the group concerto changes (or otherwise), one can utilize supplementary GKA protocols to obtain a new key. Thus, We are well-matched to the key establishment needs of self-motivated peer-to-peer networks as in ad hoc networks. While many of the future GKA protocols are too luxurious to be employed by the inhibited devices often present in ad hoc networks, others lack a formal security analysis. In this work We present a simple, sheltered and capable GKA protocol well suited to active ad hoc networks. We also present results of our accomplishment of the protocol in a prototype application. Storage outsourcing is a rising trend which prompts a number of interesting security issues, many of which have been broadly investigated in the past. However, Provable Data Possession (PDP) is a topic that has only recently appeared in the research literature. The main issue is how to recurrently and strongly verify the storage Data Server is authentically storing its client's (potentially very large) outsourced data. The storage Data Server is tacit to be untrusted in terms of both security and reliability. (In other words, it might spitefully or accidentally erase hosted data; it might also transfer it to slow or off-line storage.) The problem is exacerbated by the client being a small computing device with insulting resources. Prior work has addressed this problem using either public key cryptography or requiring the client to indenture out its data in encrypted form. A. Beimel et al has projected Social Network Services (OSNs) are one of the most popular interactive medium for communication, sharing, and disseminating a significant amount of human subsistence information. It involves swap of several types of content, including free text, picture, audio, and video data. However, OSNs features a large number of attacks that affects the privacy and security of users. So, a new community network is considered for enabling privacy security for the OSN users by offering them to open and constructive network. Here, the author uses a group key agreement problem where a user is only aware of his neighbours while the connectivity graph is random. Key agreement is a method that allows two or more parties to firmly share a secret key. R. Blom et al has proposed an ever-growing need for "trusted" devices, and as a result, assurances from technology for fiddle resistance and read-proofing of secrets stored in such devices. We converse a simple security policy - decrypt only when obligatory (DOWN) - and its inference on the security of secrets stored in trusted computers. The DOWN policy used in combination with the budding paradigm of physical un-clonable functions (PUF) can be a hopeful approach for recognition of trusted computers. A simple security policy, DOWN, can considerably improve the ability of trusted computers to protect their secrets. The emerging paradigm of physical un-clonable functions (PUF), used in concurrence with the DOWN policy, can further improve trustworthiness of computers. The need for the DOWN policy stems out of the awareness that while it may be able to protect the secrets stored in trusted computers when the computer is in-use, and when the computer is at-rest, the most susceptible period is perhaps the transition period – when the computer goes from in-use to rest state. The DOWN policy seeks to avoid unambiguous transitions.D. Boneh et al has proposed that Data storage is now an important development inclination in information technology. However, information security has become an important problem to hamper it for commercial application, such as data discretion, integrity, and accessibility. In this paper, We propose elected verifier provable data possession (DV-PDP). In public clouds, DV-PDP is a matter of critical importance when the client cannot execute the remote data possession checking. We study the DV-PDP system security model and utilize ECC-based homomorphism authenticator to design DV-PDP scheme. The scheme detached exclusive bilinear computing. Moreover in DV-PDP scheme, the Data storage Data Server is stateless and independent from verifier, which is a vital secure property in PDP schemes. Through security study and recital analysis, our scheme is demonstrable secure and high competence.D. Boneh et al has proposed Provable data possession (PDP) is a performance for ensuring the truthfulness of data in storage outsourcing. In this paper, the author address the construction of an efficient PDP scheme for distributed Data storage to support the scalability of service and data migration, in which We consider the continuation of multiple Data service providers to cooperatively store and maintain the clients' data. At hand a cooperative PDP (CPDP) scheme based on homomorphism confirmable response and hash index hierarchy. We prove the security of our scheme based on multi-prover zero-knowledge proof system, which can satisfy completeness, knowledge soundness, and zero-knowledge properties. In addition, We coherent concert optimization mechanisms for our scheme and in meticulous present a competent method for selecting optimal parameter standards to minimize the totaling costs of clients and storage service providers. Our experiments show that our explanation

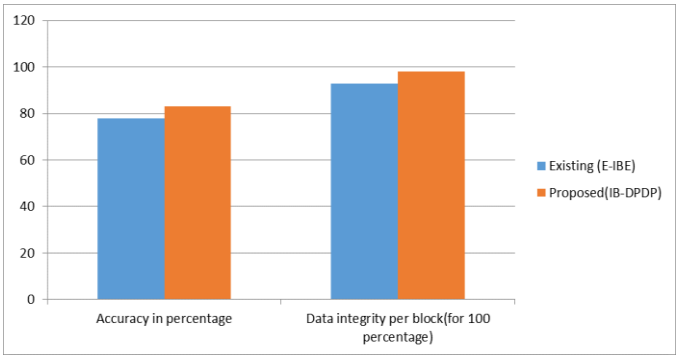
introduces lower working out and communication operating expense in comparison with non-cooperative approaches.
D. Boneh et al has proposed a fully collusion resistant tracing traitors system with sub linear size cipher texts and constant size private keys. More precisely, let N be the total number of users. Our system generates cipher texts of size O(√ N) and private keys of size O(1). We first introduce a simpler primitive We call private linear broadcast encryption (PLBE) and show that any PLBE gives a tracing traitors system with the same parameters. We then show how to build a PLBE system with O(√ N) size cipher texts. Our system uses bilinear maps in groups of composite order. D. Boneh et al proposed many storage systems rely on imitation to augment the availability and durability of data on untrusted storage systems. At present, such storage systems provide no strong confirmation that multiple copies of the data are actually stored. Storage Data Servers can conspire to make it look like We are storing many copies of the data, whereas in authenticity We only store a single copy. We address this shortcoming through multiple-replica attestable data possession (MR-PDP): A provably-secure scheme that allows a client that provisions replicas of a file in a storage system to authenticate through a challenge-response etiquette that each unique replica can be shaped at the time of the challenge and that the storage system uses t times the storage required to store a single replica. MR-PDP extends previous work on data ownership proofs for a single copy of a file in a client/Data Server storage system. Using MR-PDP to lay up t replicas is computationally much more resourceful than using a single-replica PDP format to store t separate, dissimilar files (e.g., by encrypting each file separately prior to storing it). Another benefit of MR-PDP is that it can generate further replicas on demand, at little expense, when some of the accessible replicas fail. D. Boneh et al has proposed planned cryptologic multi linear maps are terribly helpful in cryptography however their construction is one among the long-standing open drawback. Recently, 2 candidates of the cryptologic multi linear map are planned from plan of the somewhat homomorphic secret writing theme. During this speak; We have a tendency to review the definition of cryptologic multi linear map that starts with linear map with several attention-grabbing applications. We have a tendency to gift a summary of the planned 2 candidates for the multi linear map and compare their structures with the underlying somewhat homomorphic encryptions.C. Blundo et al proposed has planned Minimizing quality of cluster key exchange (GKE) protocols is a crucial milestone towards their sensible preparation. A remarkable approach to attain this goal is to alter the planning of GKE protocols by victimization generic building blocks. We have a tendency to investigate the chance of foundation GKE protocols supported a primitive referred to as multi key encapsulation mechanism (mKEM) and describe blessings and limitations of this approach. especially, We have a tendency to show the way to style a one-round GKE protocol that satisfies the classical demand of attested key exchange (AKE) security, nonetheless while not forward secrecy. As a result, We have a tendency to get the primary one-round GKE protocol secure within the customary model. In distinction to previous models We have a tendency to show the way to model each outsider and business executive KCIR among the definition of mutual authentication. Their analysis to boot implies that the business executive security compiler by Katz and Shin from ACM CCS 2005 may be wont to come through quite what's shown within the original work, particularly each outsider and business executive KCIR.

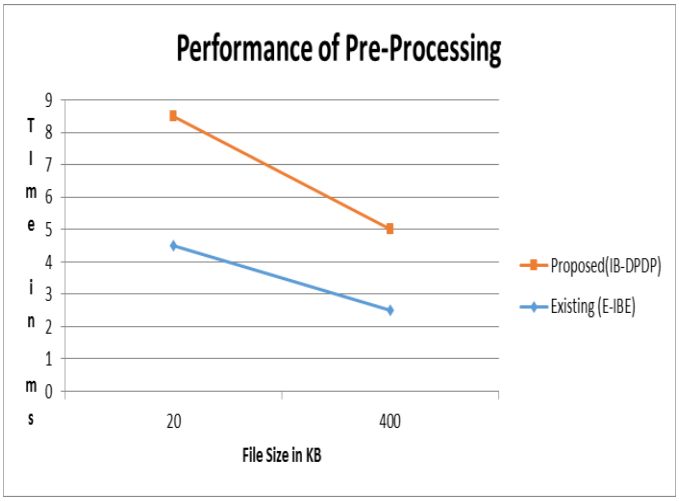
F. SYSTEM METHODOLOGY

This work mainly describes about the methods and algorithms, which are used for providing the high end of security in Data Cloud Server system and accessing data effectively and securely. On surveying the different previous works, We analyzed the advantages and disadvantages of each work and finally We derived the new technique, which over comes the drawbacks of previous work by analyzing all the information's in all state of exploration and by providing the more secured Data Cloud Server environment. Finally We conclude that our scheme provides authority that is responsible for attribute management and key distribution. For proactive broadcasting it need dynamic linking. So java will be more suitable for platform independence and networking concepts. Now a day's IT Infrastructure is propelling towards Data Cloud Server computing, but the data integrity concerns with identity privacy which must be addressed. The reviewed various privacy preserving mechanisms for static group in Data Cloud Server computing and propose a new idea for identity privacy with efficient user revocation in Data Cloud Server computing environment. Presently this research is under development to find the system for preserving identity privacy for revocation of the user or group member while sharing the data on Data Cloud Server scheme, there is an authority that is responsible for attribute management and key distribution. The authority can be the registration office in a university, the human resource department in a company, etc. The data owner defines the access policies and encrypts data according to the policies. Each user will be issued a secret key reflecting its attributes. A user can decrypt the data only when its attributes satisfy the access policies. We first propose a revocable multi authority scheme, where an efficient and secure revocation method is proposed to solve the attribute revocation problem in the system.

1. We modify the framework of the scheme and make it more practical to Data Cloud Server storage systems, in which data owners are not involved in the key generation.
2. We greatly improve the efficiency of the attributer vocation method.

.





Algorithm	Time in ms	File size in kb
Existing (E-IBE)	4.5	2.5
Proposed(IB-DPDP)	4.0	2.5

Table: 1 Preprocessing accuracy comparison between existing and proposed work

Algorithm	Over all Accuracy in percentage	Data integrity per block(for 100 percentage)
Existing (E-IBE)	78	93
Proposed(IB-DPDP)	83	98

Table: 2 Overall Accuracy and Data integrity comparison between E-IBE with Proposed IB-DPDP

IV. RESULTS

Comparing results when data are on disk versus in cache shows that disk throughput bounds IB-DPDP's performance when accessing all blocks. With the exception of the first blocks of a file, I/O and the challenge computation occur in parallel. Thus, IB-DPDP generates proofs faster than the disk can deliver data: 1.0 second versus 1.8 seconds for a 64 MB file. Because I/O bounds performance, no protocol can outperform IB-DPDP by more than the startup costs. While faster, multiple-disk storage may remove the I/O bound today. Over time increases in processor speeds will exceed those of disk bandwidth and the I/O bound will hold. Sampling breaks the linear scaling relationship between time to generate a proof of data possession and the size of the file. At 99% confidence, IB-DPDP can build a proof of possession for any file, up to 64 MB in size in about 0.4 seconds. Disk I/O incurs about 0.04 seconds of additional runtime for larger file sizes over the in-memory results. Sampling performance characterizes the benefits of IB-DPDP. Probabilistic guarantees make it practical to use public-key cryptography constructs to verify possession of very large data sets. Table 1 and 2 shows the preprocessing accuracy and overall accuracy of the proposed and existing system.

V. CONCLUSION

We have presented a decentralized access control technique with anonymous authentication, which provides user revocation and prevents replay attacks. The cloud does not know the identity of the user who stores information, but only verifies the user's credentials. Key distribution is done in a decentralized way. One limitation is that the cloud knows the access policy for each record stored in the cloud. In future, we would like to hide the attributes and access policy of a user.if an untrusted server stores a client's data. This method introduced a model for provable data possession, in which it is desirable to minimize the file block accesses, the computation on the server, and the client-server communication. The prescribed solution for PDP fits this model. Experiments show that this scheme, which offer a probabilistic possession guarantee by sampling the server's storage, make it practical to verify possession of large data sets. Previous schemes that do not allow sampling are not practical when PDP is used to prove possession of large amounts of data. This experiment shows that such schemes also impose a significant I/O and computational burden on the server.

REFERENCE

[1]Y. Amir, Y. Kim, C. Nita-Rotaru, and G. Tsudik, "On the Inf. Syst.Secur., vol. 7, no. 3, pp. 457–488, Aug. 2004.
[2] D. Augot, R. Bhaskar, V. Issarny, and D. Sacchetti, "An efficient group key agreement protocol for ad hoc networks," in Proc. 6thIEEE International Symposium, World Wireless Mobile Multimedia Networks, 2005,pp. 576–580.
[3] A. Beimel and B. Chor, "Communication in key distribution schemes," in Proceedings. Advances in Cryptology, 1994, vol. 773, pp. 444–455.
[4] R. Blom, "An optimal class of symmetric key generation systems, "in Proceedings Advances in Cryptology, 1984, vol. 209, pp. 335–338.
[5] D. Boneh and M. K. Franklin, "An efficient public-key traitor tracing scheme," in Proceedings Advances in Cryptology, 1999, vol. 1666, pp. 338–353.
[6] D. Boneh, C. Gentry, and B. Waters, "Collusion resistant broadcast encryption with short cipher texts and private keys," in Proceedings Advances in Cryptology, 2005, vol. 3621, pp. 258–275.
[7] D. Boneh, A. Sahai, and B. Waters, "Fully collusion resistant traitor tracing with short cipher texts and private keys," in Proceedings, 25th International Conference of Theory Application Cryptographic Technology, 2006, vol. 4004, pp. 573–592.
[8] D. Boneh and M. Naor, "Traitor tracing with constant sizeCipher text," in Proc. 15th ACM Conference Computer Communication Security, 2008,pp. 501–510.

[9] D. Boneh and A. Silverberg, "Applications of multi linear forms to cryptography," Contemporary Mathematical, vol. 324, pp. 71–90, 2003.
[10] C. Blundo, L. A. Mattos, and D. R. Stinson, "Generalized Beimel-Chor schemes for broadcast encryption and interactive key distribution, " theoretical. Computer Science, vol. 200, no. 1–2, pp. 313–334, 1998.

M.K.Nivodhin is an Assistant Professor in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. She received her Master of Engineering degree in Computer Science and Engineering in 2013 from Anna University, Chennai, India. She has published more than 4 papers in referred journals and conference proceedings. Her research interest includes Data Mining, Big Data, Cloud computing. She is a professional member of ISTE.

P.Vasuki is an Assistant Professor in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. She received her Master of Engineering degree in Computer Science and Engineering in 2012 from Anna University, Chennai, India. She has published more than 3s papers in referred journals and conference proceedings. Her research interest includes Big Data, Cloud computing. She is a professional member of ISTE.

K.Kiruthika is a final year student in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. Currently she is doing her final year project in security in cloud storage system.

P.N.Lokesh is a final year student in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. Currently he is doing his final year project in security in cloud storage system.

C.Sangeetha is a final year student in the Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. Currently she is doing her final year project in security in cloud storage system.

S.Suriya Kumar is a final year student in Department of Computer Science and Engineering, K.S.R. College of Engineering (Autonomous), India. Currently he is doing his final year project in security in cloud storage system.