A FULLY ENCRYPTED FRAMEWORK FOR CLOUD DATA AT BOTH CLIENT AND PROVIDER ENVIRONMENTS

Ms. Katta yamini

Master of Computer Applications, Qis College of Engineering and Technology, Ongole

Abstract: *Individuals embrace the extraordinary intensity of cloud computing, be that as it may, can't completely believe the cloud suppliers to have security delicate information, because of the nonattendance of client to-cloud controllability. To guarantee classification, information proprietors re-appropriate encoded information of plaintexts. To impart the encoded documents to different clients, Ciphertext-Policy Attribute-based Encryption (CP-ABE) can be used to lead fine-grained and proprietor driven access control. In any case, this does not adequately turned out to be secure against different assaults. Numerous past plans did not give the cloud supplier the ability to confirm whether a downloader can unscramble. In this manner, these documents ought to be accessible to everybody open to the cloud capacity. A noxious aggressor can download a large number of records to dispatch Economic Denial of Sustainability (EDoS) assaults, which will to a great extent devour the cloud asset. The payer of the cloud administration bears the cost. Plus, the cloud supplier serves both as the bookkeeper and the payee of asset utilization expense, coming up short on the straightforwardness to information proprietors. These worries ought to be settled in certifiable open distributed storage. In this paper, we propose an answer for secure encoded cloud stockpiles from EDoS assaults and give asset utilization responsibility. It employments CP-ABE conspires in a discovery way and follows discretionary access strategy of CP-ABE. We present two conventions for diverse settings, trailed by execution and security examination.*

Keywords: **Cloud Computing, Classification, public cloud storage.**

I. INTRODUCTION

Distributed storage has numerous benefits, for example, constantly on the web, pay-as-you-go, and shabby [1]. Amid these years, more information is re-appropriated to open cloud for persevering stockpiling, including individual and business documents. It conveys sasecurity worry to information proprietors [2] – [4]: the open cloud isn't trusted, and the re-appropriated information ought not to be spilled to the cloud supplier without the consent from information proprietors numerous capacities.

Frameworks use server-overwhelmed get to control, similar to secret key based [5] and certificate-based verification [6]. They excessively trust the cloud supplier to secure their delicate information. The cloud suppliers and their representatives can peruse any archive paying little heed to information proprietors' entrance arrangement. In addition, the cloud supplier can overstate the asset utilization of the file stockpiling and charge the payers more without giving verifiable records [2], [7], [8], since we come up short on a framework for verifiable calculation of the asset usage. Relying on the current server-ruled access control isn't verified. Information proprietors who store files on cloud servers still need to control the entrance without anyone else hands and keep the information confidential against the cloud supplier and malignant clients. Encryption isn't sufficient. To include the confidentiality ensure, information proprietors can scramble the files and set an entrance strategy with the goal that just qualified clients can decode the report. With Ciphertext-Policy Attribute-based Encryption (CP-ABE) [9], [10], we can have both fine-grained get to control and solid confidentiality [11] – [16]. Be that as it may, this entrance control is accessible for information proprietors, which ends up being insufficient. On the off chance that the cloud supplier can't validate clients before downloading, as in many existing CP-ABE distributed storage frameworks [14], [15], the cloud needs to enable everybody to download to guarantee accessibility. This makes the capacity framework powerless against the asset fatigue assaults. In the event that we settle this issue by having information proprietors confirm the downloaders previously enabling them to download, we lose the flexibility of access control from CP-ABE. Here records

the two issues ought to be tended to in our work: Problem I: asset fatigue assault. In the event that the cloud can't do cloud-side access control, it needs to permit anybody, including malignant aggressors, to uninhibitedly download [18], albeit just a few clients can decode. The server is powerless against asset depletion assaults. At the point when noxious clients dispatch the DoS/DDoS assaults to the distributed storage, the asset utilization will increment. Payers (in pay-as-you-go show) need to pay for the in wrinkled utilization contributed by those assaults, which is an impressive, a UN reason capable financial trouble. The assault has been presented as Economic Denial of Sustainability (EDoS) [17] – [20], which implies payers are financially assaulted in the long run. Likewise, even files are scrambled, unapproved downloads can decrease security by bringing accommodation to offline examination and spilling data like file length or refresh recurrence [19]. Issue II: asset utilization responsibility. In the compensation as-you-go show, clients pay cash to the cloud supplier for capacity administrations. The expense is chosen by asset utilization. Be that as it may, CP-ABE based plans for distributed storage get to control do not make online confirmations to the information proprietor before downloads. It is required for the cloud specialist organization to demonstrate to the payers about the real asset use. Something else, the cloud supplier can charge more without being found [21], [22].A. Rundown of Challenges and Approaches Challenge I: demonstrating the cloud supplier. Many existing CP-ABE based plans [11], [12], [23] display the cloud providers (like Google, Amazon, Microsoft Azure) as semihonest enemies or latent aggressors. Nonetheless, such a dentition is confined and it rejects some conceivable assaults in reality, for example, misrepresented asset use. To model such assaults, we consider a less confined security display, secret foe, for the cloud supplier [24]. Practically speaking, the cloud administrations are normally given by some huge IT endeavors like Google, Amazon, and Microsoft. They have to keep up great notoriety and guarantee secure distributed storage administrations to their clients. On the off chance that any endeavor the cloud supplier strays from the convention should be gotten with a probability (for example $p = 0.001$), the cloud supplier challenges not to cheat [24], [25]. Since being gotten won't just disregard the administration contracts, yet in addition lead to media presentation and pulverizes the notoriety. Mindful of the result, the cloud supplier needs to avoid assaulting, as the deceiving can be recognized. This model, undercover security, has been utilized in many secure frameworks [26], [27]. Note that the secretive security show is distinctive with the semi-legit demonstrate. The semi-fair model, which is broadly utilized in intermediaries and cloud suppliers, is a model that dwells among "vindictive" and "trusted". It displays a gathering that watches all information, yet it never executes the wrong program. Such a gathering won't cheat by definition, regardless of whether no different gatherings can identify its duping. The clandestine model, which lives among "malignant" and "semi-genuine", models this gathering in an unexpected way. It won't execute the wrong program just if there is an instrument to distinguish its swindling. On the off chance that no location exists in the framework, the gathering may even trade off the information, for instance. Hence, it is increasingly handy for open distributed storage. Approach: show cloud suppliers as undercover foes, and plan conventions strong to a secret enemy. Test II: perfect with existing frameworks. There are numerous developments and variations for CP-ABE [23], [28], [29]. We don't plan another variation of CP-ABE to determine the first challenge, as it is difficult to accomplish every one of the functionalities in these frameworks and furthermore it's a bit much. Other than the functionalities, a few variations give extra security and protection ensure. For instance, the literary works [12], [16] conceals the entrance arrangement. On the off chance that the cloud-side access control makes the cloud supplier knowing the entrance strategy, it isn't viewed as secure and good. It requires the cloud side access control to be zero-knowledge for arbitrary CP-ABE plans. Approach: use CP-ABE in a grammatical and discovery way and guarantee the development not spilling strategy and properties. The framework just realizes whether the client is genuine or not, and that's it. Test III: negligible execution overhead. To secure the distributed storage successfully against the asset depletion assault, the cloud-side access control should be efficient and lightweight, generally if the cloud server spends, for instance 20ms [30], executing the cloud-side access control, it will end up being a computational asset fatigue assaults, which can be utilized by malignant assailants for DDoS and EDoS. The execution overhead being little additionally benefits the information clients who download the files from the distributed storage, makingthe calculation not congenial to asset restricted gadgets [31]. Approach: plan an efficient get to control for the cloud supplier which ought not to include an excess of overhead.B. Our work and Contribution In this paper, we consolidate the cloud-side access control and the current information proprietor side CP-ABE based access control [32], to determine the previously mentioned security issues in privacypreserving distributed storage. Our strategy can keep the EDoS assaults by giving the cloud server the capacity to check whether the client is approved in CP-ABE based plan, without releasing other data. For our cloud-side access control, we use CP-ABE encryption/decoding amusement as test reaction. While transfer an encoded file, the information proprietor firstly creates some irregular test plaintexts and the comparing ciphertexts [33]. The ciphertexts are identified with a similar access strategy with the specific file. For an approaching information client, the cloud server asks him/her to unscramble arbitrarily chosen test ciphertext. On the off chance that the client demonstrates a right outcome, which implies he/she is approved in CP-ABE, the cloud-side access control permits the file download. To make our answer secure and efficient in certifiable applications, we give two conventions of cloud-side and information proprietor side joined access control. The primary commitment of this work can be outlined as pursues. 1) We propose a general answer for secure encoded distributed storage to keep the EDoS

assaults, just as have fine-grained get to control and asset utilization responsibility. To the best of our insight, this is the first work to guarantee that insufficient cloud-side access control in encoded distributed storage will prompt EDoS assaults and gives a down to earth arrangement. The arrangement can be good with numerous CP-ABE plans. 2) For various information proprietor online examples and execution concern, we give two conventions to validation and asset utilization bookkeeping. We likewise present the blossom filter and the probabilistic check to improve the efficiency yet at the same time ensure the security. 3) Compared with many condition of-expressions developments of scrambled distributed storage that accept the presence of a semihonest cloud supplier, we utilize a progressively reasonable risk show where we expect the cloud supplier to be a clandestine foe [24], which gives higher security ensure.

II. RELATED WORKS

To conduct a fine-grained knowledge owner-side access management in public cloud storage, that is semi-honest, Attribute-based Encryption (ABE) [9], [30], [31] is introduced [23]. Among ABE schemes, CP-ABE [9], [10] is sensible publically cloud storage, within which the ciphertext is encrypted beneath Associate in Nursing access policy and solely users whose attributes satisfy the access policy can decode the ciphertext. Afterward, several variants and relevant protocols [14], [16], [32], [33] are projected to make CP-ABE a lot of appropriate for real eventualities with wealthy functionalities and security properties publically cloud storage. The cryptography-driven access management doesn't defend the cloud supplier against several different attacks. Since the cloud provider doesn't conduct the access management, it cannot stop those unauthorized users. One attack that's originated from this limitation is Distributed Denial of Services (DDoS). The power of DDoS attacks has been showed to incur important resource consumption in computer hardware, memory, I/O, and network [34]. The attacks will exist publically clouds [17], the limitation of cloud-side static resource allocation model is analyzed, together with the danger of Economic Denial of property (EDoS) attacks, that is that the case of DDoS attacks within the cloud setting or the deceitful Resource Consumption (FRC) attack in [17]. These attacks area unit supposed to break the budget of public cloud customers. Some existing works attempt to mitigate EDoS attacks [19], in [19], the authors projected a mitigation technique by confirmative whether or not a request comes from a cloud user or is generated by bots. In [33], the authors projected Associate in nursing attribute-based thanks to determine malicious shoppers. They treat the underlying application in recording equipment and don't absolutely immunize the attack within the algorithmic and protocol level. Some existing works discuss the required of accounting resource consumption within the public cloud arouses some considerations [7], [8], [21], [22]. Within the literature [21], the authors discussed key problems and challenges concerning a way to bring home the bacon accountability in cloud computing. Within the literature [22], the authors surveyed existing accounting and responsibility in content distribution architectures. Within the literatures [7] and [8], the authors severally projected a scientific approach for verifiable resource accounting in cloud computing. However, the accounting approach involves changes to the system model, and needs the anonymous verification of users, which is not supported in previous systems. Compared with relevant schemes, our approach works on the protocol level to supply the resource verifiability that depends on licensed users World Health Organization satisfy the CP-ABE policy, and achieves the covert security which is a lot of sensible and secure.

III. LITERATURE SURVEY

"Secret-Sharing Schemes: A Survey,"

A secret-sharing scheme is a method, by which a dealer distributes shares to parties such that only authorized subsets of parties can reconstruct the secret. Secret-sharing schemes are important tools in cryptography and they are used as a building box in many secure protocols, e.g., general protocol for multiparty computation, Byzantine, agreement, threshold cryptography, access control, attribute-based encryption, and generalized oblivious transfer. In this survey, we will describe the most important constructions of secret-sharing schemes, explaining the connections between secret-sharing schemes and monotone formulae and monotone span programs. The main problem with known secret-sharing schemes is the large share size: it is exponential in the number of parties. We conjecture that this is unavoidable. We will discuss the known lower bounds on the share size. These lower bounds are fairly weak and there is a big gap between the lower and upper bounds. For linear secret-sharing schemes, which is a class of schemes based on linear algebra that contains most known schemes, super-polynomial lower bounds on the share size are known. We will describe the proofs of these lower bounds. We will also present two results connecting secret-sharing schemes for a Hamiltonian access structure to the NP vs. coNP problem and to a major open problem in cryptography – constructing oblivious-transfer protocols from one-way functions.

Using Erasure Codes Efficiently for Storage in a Distributed System

Erasure codes provide space-optimal data redundancy to protect against data loss. A common use is to reliably store data in a distributed system, where erasure-coded data are kept in different nodes to tolerate node failures without losing data. In this paper, we propose a new approach to maintain ensure-encoded data in a distributed system. The approach allows the

use of space efficient k-of-n erasure codes where n and k are large and the overhead n-k is small. Concurrent updates and accesses to data are highly optimized: in common cases, they require no locks, no two-phase commits, and no logs of old versions of data. We evaluate our approach using an implementation and simulations for larger systems.

Security amplification by composition: The case of doublyiterated, ideal ciphers

One concern in using cloud storage is that the sensitive data should be confidential We investigate, in the Shannon model, the security of constructions corresponding to double and (two-key) triple DES. That is, we consider Fk1 (Fk2 ()) and Fk1 (F 1 k2 (Fk1 ())) with the component functions being ideal ciphers. This models the resistance of these constructions to \generic" attacks like meet in the middle attacks. sense. We compute a bound on the probability of breaking the double cipher as a function of the number of computations of the base cipher made, and the number of examples of the composedcipher seen, and show that the success probability is the square of that for a single key cipher. meet in the middle is the best possible generic attack against the double cipher. local revocable group signature and identity-based broadcast encryption with constant size ciphertext and private keys. To realize our concept, we equip the broadcast encryption with the dynamic ciphertext update feature, and give formal security guarantee against adaptive chosen-ciphertext decryption and update attacks.

IV. EXISTING SYSTEM

Some existing works try to mitigate EDoS attacks. In the authors proposed a mitigation technique by verifying whether a request comes from a cloud user or is generated by bots. The authors proposed an attribute based way to identify malicious clients. They treat the underlying application in a black box and do not fully immunize the attack in the algorithmic and protocol level.

IV. PROPOSED SYSTEM

To achieve the security requirements, the scheme consists of two componetns:

1) A cloud-side access control to block users whose attribute set Ai does not satisfy the access policy A;

2) A proof-collecting subsystem where the cloud provider can collect the proofs of resource consumption from users, and present to the data owners later. In real-world scenarios, it is reasonable to specify an expected maximal download times, and data owners can remain offline unless it wants to increase this value. This leads to our first protocol: Partially Outsourced Protocol (POP). In some other cases where the data owner cannot set an expectations of download times or would be offline for a long time, the data owner can delegate to the cloud. This leads to our second protocol: Fully Outsourced Protocol (FOP).

V. SYSTEM MODEL

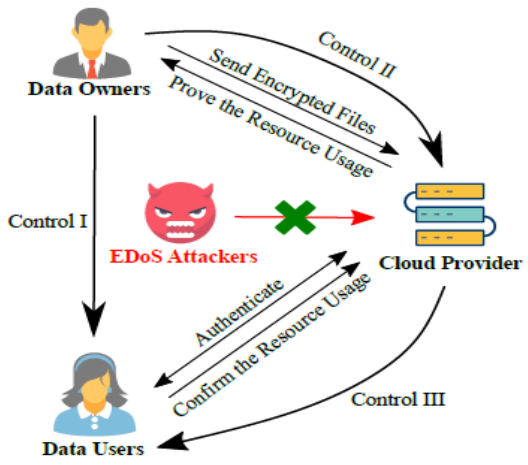
As shown in Fig. 1, the cloud storage system consists of three entities: information house owners, data users, and therefore the cloud supplier. information house owners ar the owner and publisher of files and pay for the resource consumption on file sharing. As the payers for cloud services, the info house owners need the transparency of resource consumption to make sure honest billing. the info house owners need the cloud supplier to justify the resource usage. In our system, the info owner is not forever on-line. information users need to get some files from the cloud provider keep on the cloud storage. they have to be authenticated by the cloud supplier before the transfer (to thwart EDoS attacks). The approved users then confirm (and sign for) the resource consumption for this download to the cloud supplier. Cloud supplier hosts the encrypted storage and is usually online. It records the resource consumption and charges data house owners supported that record. The cloud isn't public-accessible in our system because it has Associate in Nursing authentication based access management. solely information users satisfying the access policy will transfer the corresponding files.

The cloud supplier additionally collects the proof of the resource consumption to justify the charge. As shown in Fig. 1, we've 3 controls among 3 entities in our system:

management I. information house owners assign Associate in Nursing access policy within the document, that controls the set of knowledge users WHO have the privileges to decode the contents. management II. information house owners verifies the resource consumption from the cloud supplier, that controls the cloud provider to not exaggerate the resource usage. management III. The cloud supplier verifies whether or not the user will decode before the transfer, that controls the ability of a malicious user WHO launches DDoS/EDoS attacks.

Moreover, our system differs from previous cloud storage constructions, as we tend to take under consideration the resource consumption.

In observe, the cloud services ar sometimes charged according to the resource consumption, which has the resource spent on attackers. The DDoS/EDoS attacks can invariably succeed and lift the overhead, that is controlled in our system because of the introduction of the cloud-side access control.



Sufficient Detection Rate from Probabilistic Check The probabilistic check reduces the overhead of verification and maintains the covert security. think about the cloud supplier forges a proportion of proofs and also the knowledge owner checks with the proportion . The detection risk is as follows:

$$p = \begin{cases} 1 & \text{if } \beta \geq 1-\alpha, \\ 1 - \binom{n(1-\alpha)}{n\beta} / \binom{n}{n\beta} & \text{if } \beta < 1-\alpha. \end{cases}$$

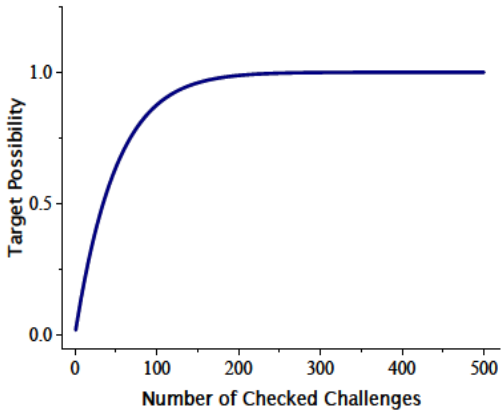
If the cloud forges one invalid proof, the possibility of being detected is at least:

$$p \geq 1 - \binom{n}{(n+1)\beta} / \binom{n+1}{(n+1)\beta} = \beta = O(1),$$

which is ample for convert security if hen the information owner selects a non-negative constant (like zero.001). To demonstrate however probabilistic check reduces the information measure, we offer associate example: Suppose there ar N = one thousand challenges and therefore the cloud needs to forge N0 = twenty (= 0:02) proofs. we have a tendency to choose completely different (N + N0) and calculate a possibility ptarget catching the misbehaviors:

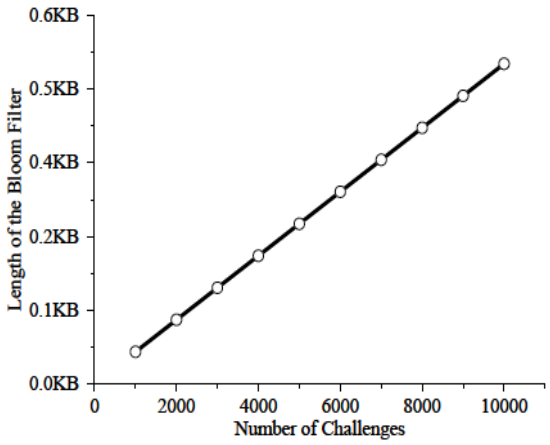
$$1 - \binom{N}{(N+N')\cdot\beta} / \binom{N+N'}{(N+N')\cdot\beta} \geq p^{\text{target}}.$$

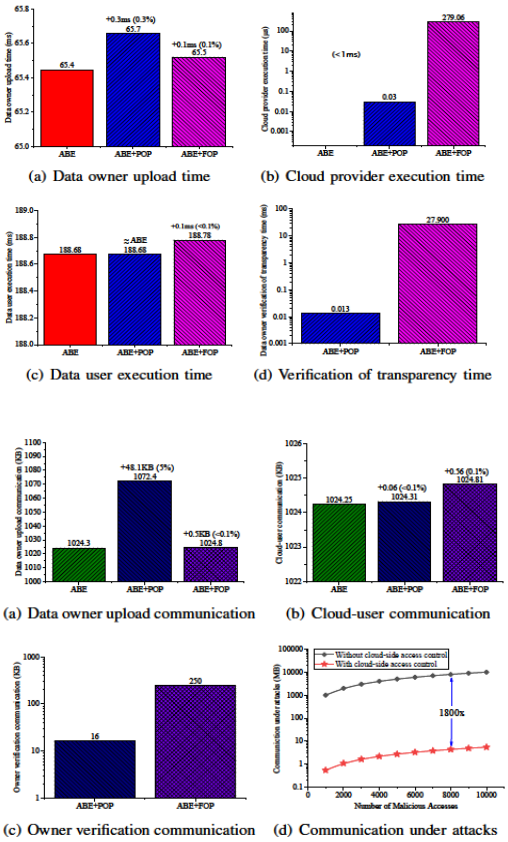
This result's shown. From the definition of covert security, having a detection likelihood of 100 percent is ample to force the cloud supplier t behave, whereas setting = 0:1% (checking one out of 1020 proofs) has a pair of catching chance. Small increase of will result in high catching chance as = 0:6% (checking vi out of 1020 proofs) has 100 percent catching possibility. If the cloud needs to get several pretend proofs to charge considerably extra money, the catching chance increases.



For reducing the storage of challenge plaintexts in POP, we use the bloom filter. we would like to indicate the false positive rate is enough in order that secure against covert security. Assume the number of hash functions is l = m n ln a pair of, and the proportion m=n could be a non-negative constant, we have:

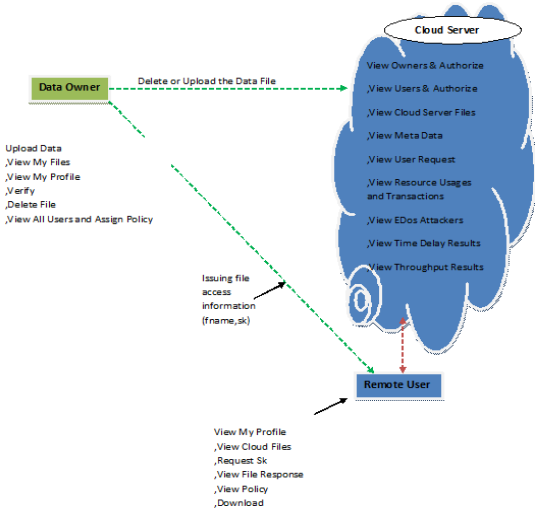
$$fp \approx (0.5^{\ln 2})^{\gamma} > \frac{1}{2\gamma} > 0.$$





VI. METHODOLOGY

The proposed system architecture and its components are given by



Data Owners

Data owners are the owner and publisher of files and pay for the resource consumption on file sharing. As the payers for cloud services, the data owners want the transparency of resource consumption to ensure fair billing. The data owners require the cloud provider to justify the resource usage. In our system, the data owner is not always online.

Data Users

Data users want to obtain some files from the cloud provider stored on the cloud storage. They need to be authenticated by the cloud provider before the download (to thwart EDoS attacks). The authorized users then confirm (and sign for) the resource consumption for this download to the cloud provider.

Cloud Server

Cloud provider hosts the encrypted storage and is always online. It records the resource consumption and charges data owners based on that record. The cloud is not public-accessible in our system as it has an authentication based access control. Only data users satisfying the access policy can download the corresponding files. The cloud provider also collects the proof of the resource consumption to justify the billing.

VII. RESULT

Security algorithms mentioned for encryption and decryption can be implemented in future to enhance security framework over the network. In the future, I will try to develop algorithm to make advancement to my research by providing algorithm for encryption, decryption and batch auditing to provide authentication.

VIII. CONCLUSION

We propose a combined the cloud-side and knowledge owner-side access management in encrypted cloud storage, that is proof against DDoS/EDoS attacks and provides resource consumption accounting. Our system supports discretionary CP-ABE constructions. the development is secure against malicious knowledge users and a covert cloud supplier. we tend to relax the safety demand of the cloud supplier to covert adversaries, that could be a a lot of sensible and relaxed notion than that with semi-honest adversaries. to create use of the covert security, we tend to use bloom filter and probabilistic sign in the resource consumption accounting to scale back the overhead. Performance analysis shows that the overhead of our construction is little over existing systems.

IX. REFERENCES

[1] F. Cacheda, V. Carneiro, D. Fernandez, and V. Formoso, ‘‘Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems,’’ *ACM Trans. Web*, vol. 5, no. 1, p. 2, Feb. 2011.

[2] C. A. Gomez-Uribe and N. Hunt, ‘‘The netflix recommender system: Algorithms, business value, and innovation,’’ *ACM Trans. Manage. Inf. Syst.*, vol. 6, no. 4, p. 13, Jan. 2015.

[3] M. Deshpande and G. Karypis, ‘‘Item-based top-n recommendation algorithms,’’ *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 143–177, Jan. 2004.

[4] D. Gonzales, J. Kaplan, E. Saltzman, Z. Winkelman, and D. Woods, ‘‘Cloud-trust—A security assessment model for infrastructure as a ser- vice (IaaS) clouds,’’ *IEEE Trans. Cloud Comput.*, to be published, doi: 10.1109/TCC.2015.2415794.

[5] J. Heizer and B. Render, *Operations Management*, 7th ed. Upper Saddle River, NJ, USA: Pearson, 2004.

[6] K. Hwang, G. C. Fox, and J. J. Dongarra, *Distributed and Cloud Computing From Parallel Computing to the Internet of Things*, 1st ed. Waltham, MA, USA: Morgan Kaufmann, 2012.

[7] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*, 1st ed. New York, NY, USA: Cambridge Univ. Press, 2010.

[8] Y. Shi, M. Larson, and A. Hanjalic, ‘‘Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges,’’ *ACM Comput. Surv.*, vol. 47, no. 1, p. 3, Jul. 2014.

[9] L. Sun, H. Dong, F. K. Hussain, O. K. Hussain, and E. Chang, ‘‘Cloud service selection: State-of-the-art and future research directions,’’ *J. Netw. Comput. Appl.*, vol. 45, pp. 134–150, Oct. 2014.

[10] L. A. Tawalbeh, R. Mehmood, E. Benkhlifa, and H. Song, ‘‘Mobile cloud computing model and big data analysis for healthcare applications,’’ *IEEE Access*, vol. 4, pp. 6171–6180, Sep. 2016.

[11] Z. Zheng, H. Ma, M. R. Lyu, and I. King, ‘‘QoS-aware Web service rec- ommendation by collaborative filtering,’’ *ACM Trans. Services Comput.*, vol. 4, no. 2, pp. 140–152, Apr./Jun. 2011.

[12] Z. Zheng, X. Wu, Y. Zhang, M. R. Lyu, and J. Wang, ‘‘QoS ranking prediction for cloud services,’’ *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1213–1222, Jun. 2013.

[13] X. Zheng, P. Martin, K. Brohman, and L. D. Xu, ‘‘CLOUDQUAL: A qual- ity model for cloud services,’’ *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1527–1536, May 2014.

[15] EMC, “Transform to a Hybrid Cloud,” http://www.emc.com/campaign/global/hybridcloud/index.htm.

[16] IBM, “IBM Hybrid Cloud Solution,” http://www 01.ibm.com/software/tivoli/products/hybrid-cloud/.

[17] J. Kilian and P. Rogaway, “How to protect DES against exhaustive key search,” in *Advances in Cryptology (CRYPTO)*, 1996, pp. 252–267.

[18] M. Klonowski, P. Kubiak, and M. Kutylowski, “Practical Deniable Encryption,” in *Theory and Practice of Computer Science (SOFSEM)*, 2008, pp. 599–609.

[19] H. Krawczyk, “Secret Sharing Made Short,” in *Advances in Cryptology (CRYPTO)*, 1993, pp. 136–146.

[20] J. Kubiatowicz, D. Bindel, Y. Chen, S. E. Czerwinski, P. R. , D. Geels, R. Gummadi, S. C. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Y. Zhao, “OceanStore: An Architecture for Global-Scale Persistent Storage,” in *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2000, pp. 190–201.

[21] L. Lamport, “On interprocess communication,” 1985.

[22] S. Micali and L. Reyzin, “Physically observable cryptography (extended abstract),” in *Theory of Cryptography Conference (TCC)*, 2004, pp. 278–296.

[23] NEC Corp., “HYDRAstor Grid Storage,” http://www.hydrastor.com.

[24] M. O. Rabin, “Efficient dispersal of information for security, load balancing, and fault tolerance,” *J. ACM*, vol. 36, no. 2, pp. 335–348, 1989.

[25] J. K. Resch and J. S. Plank, “AONT-RS: Blending Security and Performance in Dispersed Storage Systems,” in *USENIX Conference on File and Storage Technologies (FAST)*, 2011, pp. 191–202.

[26] R. L. Rivest, “All-or-Nothing Encryption and the Package Transform,” in *International Workshop on Fast Software Encryption (FSE)*, 1997, pp. 210–218.

[27] A. Shamir, “How to Share a Secret?” in *Communications of the ACM*, 1979, pp. 612–613.

[28] D. R. Stinson, “Something About All or Nothing (Transforms),” in *Designs, Codes and Cryptography*, 2001, pp. 133– 138.

[29] StorSimple, "Cloud Storage," http://www.storsimple.com/.

[30] J. H. van Lint, *Introduction to Coding Theory*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1982.

[31] Wikipedia, "Edward Snowden," http://en.wikipedia.org/wiki/Edward_Snowden#Disclosure.

[32] Z. Wu, M. Butkiewicz, D. Perkins, E. Katz-Bassett, and H. V.Madhyastha, "SPANStore: Cost-effective Geo-replicated Storage Spanning Multiple Cloud Services," in *ACM Symposium. on Operating Systems Principles (SOSP)*, 2013, pp. 292–308.

[33] H. Xia and A. A. Chien, "RobuSTore: a Distributed Storage Architecture with Robust and High Performance," in *ACM/IEEE Conference on High Performance Networking and Computing (SC)*, 2007, p. 44.

Authors Profile

Ms. **Katta yamini** pursuing MCA 3rd year in Qis College and Engineering and Technology in Department of Master of Computer Applications, Ongole.