Quality Based Secure Data Sharing with Efficient Revocation in Fog Computing

Dr.k. Satish[1], P. Venkata Naga Vasanta[2], Sk. Afroz[3], N. Aravind[4], S. Ramya[5]
[1]*Professor, Dept of CSE, Tirumala Engineering College, Narasaropet, Guntur, A.P., India*
[2345]*B. Tech Students, Dept of CSE, Tirumala Engineering College, Narasaropet, Guntur, A.P., India*

Abstract- Fog computing is an idea that expands the worldview of distributed computing to the system edge. The objective of mist processing is to arrange assets in the region of end-clients. Likewise with distributed computing, mist computing gives stockpiling administrations. The information proprietors can store their classified information in many mist hubs, which could cause more difficulties for information sharing security. Right now, present a novel engineering for information partaking in a Fog domain. We investigate the advantages of Fog computing in tending to one-to-numerous information sharing applications. This engineering tried to beat the cloud-based design and to guarantee further upgrades to framework execution, particularly from the viewpoint of security. We will address the security difficulties of information sharing, for example, fine-grained get to control, information secrecy, intrigue opposition, adaptability, and the issue of client denial. Remembering these issues, we will verify information partaking in Fog computing by consolidating characteristic based encryption and intermediary re-encryption methods. The discoveries of this examination show that our framework has the reaction and handling time quicker than traditional cloud frameworks. Further, exploratory outcomes show that our framework has a proficient client repudiation system and that it gives high adaptability and sharing of information progressively with low inertness.

Keywords- Attribute-Based Encryption, Fine-Grained Access Control, Fog Computing, Proxy Re-Encryption, User Revocation

I. INTRODUCTION

Distributed computing is the most mainstream processing worldview that offers IoT assets over the Internet. Distributed computing gives numerous focal points to end-clients, for example, lower cost, high unwavering quality, and more prominent adaptability. Be that as it may, it has a few disadvantages, which incorporate a high inertness, requiring Internet network with high transfer speed and security [1]. During the most recent couple of years, another pattern of Internet organizations rose called the Internet of Things (IoT) that imagines having each gadget associated with the Internet. IoT applications incorporate e-medicinal services, a keen matrix, and so forth. Those applications require low idleness, versatility support, geo-dissemination, and client area mindfulness. Distributed computing has all the earmarks of being a wonderful answer for offer administrations to end-clients, however it can't meet the IoT's' prerequisites. Therefore, a promising stage called mist computing is expected to give the IoT's' prerequisites; Fog computing was proposed by Cisco in 2012 [2].

Fog computing is an idea that broadens the worldview of distributed computing to the system edge, considering another age of administrations [3]. Fog computing has a middle of the road layer situated between end gadgets and distributed computing. This prompts a model with a three-layer progressive system: Cloud-Fog-End Users [4]. The objective of Fog computing is to offer assets in closer region to the end-clients. As in Figure 1, each mist is situated at a particular structure and offers administrations to those inside the structure [4]. Mist computing underpins low idleness, client versatility, continuous applications, and wide geographic circulation. Additionally, it upgrades the nature of administrations (QoS) for end-clients. These highlights make the Fog a perfect stage for the IoT [5]. Backing for area mindfulness is the basic distinction between the cloud condition and the Fog condition. Distributed computing fills in as a concentrated worldwide model, so it needs area mindfulness. Rather than distributed computing, Fog gadgets are genuinely arranged in the region of end-clients [6]. Information sharing has incredible significance for some individuals, and it is a critical need for associations that expect to improve their efficiency [7]. Presently, there is a critical need to create information sharing applications, particularly for mass correspondences, where the information proprietor is liable for conveying shared assets to a huge gathering of clients.

This one too much technique needs unique consideration, mulling over the provokes identified with such applications. The fundamental issues for such applications are issues identified with security and protection [8]. Like distributed computing, Fog computing faces a few security dangers for information stockpiling; to meet them, there are security includes that were given in the cloud condition. These security highlights are the authorizing of fine-grained get to control, information classification, and client denial and intrigue obstruction between substances [9]. We present a novel

engineering for information sharing a Fog domain. We investigate the advantages brought by mist computing to address a one-to-numerous information sharing application. Such engineering is looked to beat the cloud-based design and guarantee further upgrades to framework execution, particularly from the point of view of security. Our proposed system gives high versatility and sharing of information continuously with low inactivity.

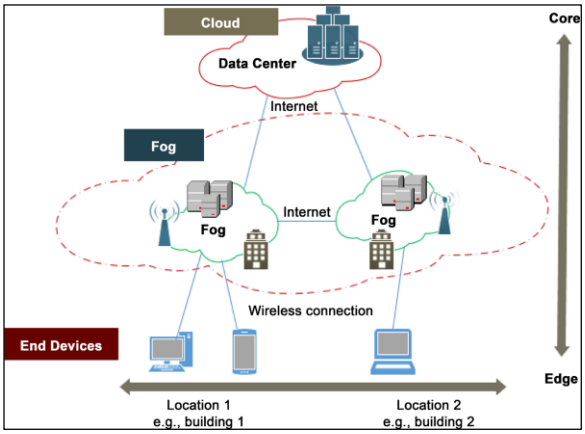

Fig.1: The fog is situated between the cloud and the edge.

II. RELATED WORK

We will give a point by point diagram of earlier examinations on secure information partaking in cloud situations. Yu et al. [9] proposed an information sharing plan intended to give fine-grained information get to control, information classification, and adaptability. Be that as it may, it requires refreshing all clients' mystery keys and re-scrambling all the records, accordingly decreasing the effectiveness of the client denial activity. Wu et al. [10] displayed a novel method for sharing media, particularly in enormous appropriated frameworks. Lamentably, the decoding activity in low-end gadgets is moderate, and client renouncement isn't tended to. Liu et al. [11] planned a structure for sharing information dependent on the time idea. It is a superior fit for a situation where in the information proprietor is disconnected, and intermittent client renouncement happens. In any case, the proposed plot requires productive shared periods for all the client related characteristics. Tu et al. [12] proposed a protected information sharing structure that is secure against picked ciphertext assaults. Sadly, the proposed system places enormous calculation overhead on the procedure of client denial.

Yang and Zhang [13] planned a nonexclusive plan for sharing information. The plan doesn't have to require the redistribution of keys. Notwithstanding, it has not tended to the situation wherein a repudiated client rejoins the gathering with new access rights. Hur [14] proposed a safe information sharing plan including quick client renouncement. IoT's significant downside is that it experiences low versatility and high calculative unpredictability. Samanthula et al. [15] proposed a system with viable client disavowal. Shockingly, the proposed plot puts a substantial weight on the cloud servers by requiring the information proprietor to make a token in each record for each client, which expands the unpredictability of the framework and decreases versatility. From the past conversation, it is apparent that the past plans have neglected to locate a general answer for accomplishing the past objectives, as appeared in Table 1. The greater part of these ideal highlights are acknowledged in [9], so we will apply it in a mist situation with some improvement to accomplish all our plan objectives. Our proposed system lays on a blend of past methodologies that give secure information partaking in distributed computing, for example, Attribute-Based Encryption (ABE) and Proxy Re-Encryption (PRE) procedures [9] [16] [17]. Not at all like the past framework [9], the proposed disavowal component doesn't require the re-encryption of all framework records and refreshing of every single mystery keys. Our proposed framework gives constant information sharing to assemble individuals. Our work will concentrate on giving a perfect domain to verify information partaking in a mist situation to conquer the disservices of a cloud-based information sharing framework, which incorporates a high inactivity, requiring Internet network with high data transmission and lacking area mindfulness.

Table 1. The main features schemes.

Design Goals	References							
	9	10	11	12	13	14	15	16
Data confidentiality	Y	Y	Y	Y	Y	Y	Y	Y
Enforcing fine grained access control.	Y	Y	Y	Y	Y	Y	Y	Y
Scalability	Y	Y	Y	Y	Y	N	N	Y
Efficient user revocation	N	N	Y	N	Y	Y	Y	N
Collusion resistance	Y	N	N	Y	N	Y	Y	Y
Real-time data sharing	N	N	N	N	N	N	N	N

III. Fog Based Data Sharing Architecture

A. Fog Based Data Sharing Model

There are four gatherings in the proposed framework: Data Owner, Cloud Servers, and many Fog Nodes and Data clients.

• Data Owner (DO) has the option to get to and modify the information. He scrambles the information with the traits of a particular gathering and produces the decoding keys for clients. At that point, he transfers the encoded information to the cloud servers.

• Cloud Server (CLD) is liable for information stockpiling and conveys the information to the Fog hubs.

• Fog Nodes (FNs) are liable for information stockpiling and for tending to clients' solicitations. They are considered as a semi-confided in party. They execute tasks of the client repudiation stage.

• Data Users (Us) are the individuals who demand information get to when they reserve the privilege to get to information.

This implies, just when the client's entrance arrangement fulfills the information qualities. The mist condition situation is appeared in Figure 2, where a DO scrambles an information record and afterward redistributes it to a CLD for capacity. At that point, the CLD conveys the information document to the particular mist hub by means of the information dissemination convention, as will be indicated later. Fog hubs are geologically dispersed inside a particular area, and they have fixed areas. The client can be moving, and he is mentioning the information from the Fog hub nearest to him. The mist hub gets the client's solicitation and conveys the record to the client. The DO can assign a large portion of the undertakings to the home Fog hubs, as appeared in the accompanying area. In a mist based information sharing model, mist hubs and the information proprietor both can be associated with the cloud by means of the Internet. The mist hubs are associated by means of a wired system over the Internet. The clients can be associated with the Fog hub utilizing a remote association procedure, for example, Wi-Fi, as appeared in Figure 2. This model comprises of gatherings of clients, and each gathering has a lot of qualities and a focal area. Each gathering has numerous clients who share similar qualities. One of the gathering ascribes alludes to IoT area, and gathering individuals interface with a Fog that has a similar area. The information proprietor appoints numerous records to each gathering dependent on the properties and requirements of IoT individuals.

Each Fog hub serves one gathering and is free in IoT activity, so it isn't influenced when a client is renounced from another gathering. Thusly, the proposed repudiation system requires the re-encryption of the influenced records and the refreshing of the mystery keys, just for one gathering.

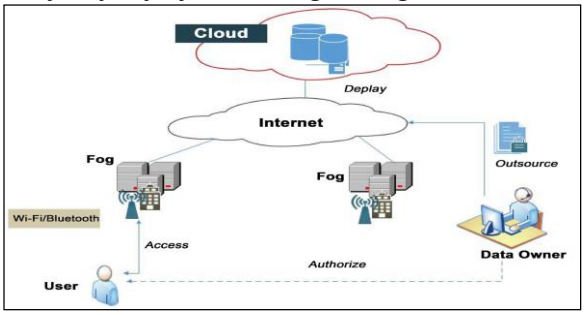

Fig.2: Fog-based data-sharing model.

B. Data Distribution Protocol

Two kinds of mist in the information circulation design are characterized:

• Home Fog (HF): the Fog has a similar area as the client's unique area, where clients are well on the way to be found. It stores the client's information and deals with the procedures.

• Foreign Fog (FF): the mist is found away from the client's unique area, where the client is presently living, as appeared in Figure 3.

• The proposed framework is involved two sorts of server farms:

• Cloud server farms (which incorporate the server farms for each gathering).

• Local mist server farms.

Each mist hub is considered the "Home Fog" for the gathering that has the part's equivalent area, while it is considered the "Outside Fog" for different gatherings. A neighborhood server farm is mist stockpiling that holds duplicates of mystery records. It is preloaded with the information required by mist clients. The mist hubs keep up correspondence with the cloud. The information sharing between the cloud server farm and each mist hub server farm is performed through prompt synchronization dependent on the unicast technique. At the point when the client demands a document from the mist hub, if the mist is the client's HF, the Fog hub straightforwardly sends the record to the client. In the event that the client is away from his/her HF, the case is handled, as appeared in Figure 4.

1) Using verification, a client logs to the Fog hub nearest to him. He demands to go along with it and recognizes the time of the joined Fog hub through the enlistment procedure.

2) The FF perceives the client's home by the framework client list (the cloud refreshes this rundown at whatever point a client is included or evacuated and sends it to all mist hubs by means of broadcasting after each update. This rundown incorporates the client's ID and IoT HF.

3) The FF sends the getting message together with the predefined period to the client's HF.

4) The HF sends an acknowledgment answer to recognize the joining.

5) The FF acknowledges the client as a guest, refreshes IoT guest rundown, and afterward synchronizes the rundown with the cloud.

6) The HF refreshes the area of IoT clients in Table 2 by changing the client's area to the FF's area and synchronizing it with the cloud. This table does exclude the guest's clients; it is just for IoT bunch individuals.

7) The HF sends the client's mystery information to the FF.

8) The FF stores the information in the IoT server farm. In the event that the time terminates and the client is still at the FF, he should join the FF once more. At the point when the client comes back to his HF, he will send a de-joined solicitation to the HF and illuminate it that he is at his HF. The FF refreshes the present area table and synchronizes the table with the cloud.

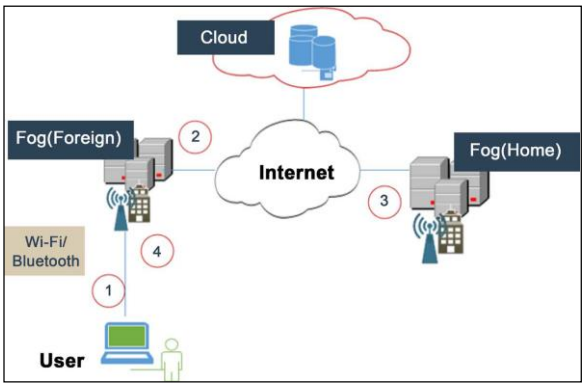

Fig.3: Data distribution architecture.

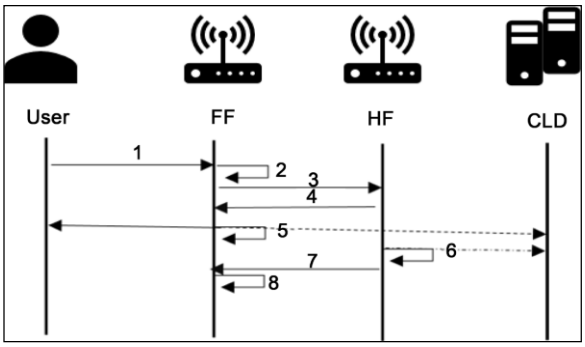

Fig.4: Data distribution protocol.

IV. THE PROPOSED SYSTEM

A. Technique Preliminaries

1) Key Policy Attribute-Based Encryption (KP-ABE)

In KP-ABE, information have a lot of credits connected to information by encryption with the open key. Every client has an entrance structure that is an entrance tree related with information qualities. The client's mystery key is an impression of the client's entrance tree; subsequently, the client can unscramble a ciphertext if the information credits to coordinate their entrance tree [13] [18].

2) Proxy Re-Encryption (PRE)

PRE is a cryptographic crude that permits a semi-confided in intermediary to change the ciphertext of the scrambled information under the information proprietor's open key into an alternate ciphertext under the gathering part's open key. The semi-confided in intermediary server needs a re-encryption key sent by the information proprietor for a fruitful transformation procedure, and it can't find the basic plaintext of the encoded information. Just an approved client can decode the ciphertext [19].

B. Design Goals

The structure objectives are as per the following: • Data secrecy: Unauthorized clients (counting the Fog and cloud servers) are not permitted to get to the information [9]. Fine-grained get to control: The information proprietor can decide the entrance structure for every client [11]. Client renouncement: Revoked clients can't reaccess the information. Versatility and effectiveness: The framework must keep up both proficiency and adaptability, in any event, when the quantity of clients increments [9]. Conspiracy opposition: disallows unapproved parties from participating so as to discover the substance of touchy information [20]

C. Assumptions and Security Models

In the proposed structure, the information sharing framework is one too much. The mist hubs have fixed areas. It might be accepted that the objective client is a PC or other cell phone. Additionally, that the information proprietor and clients have as of now people in general/private key sets, where the open keys can be anything but difficult to get by different substances. Utilizing the security conventions, the correspondence channels are verified between the information proprietor/cloud server and mist hubs, for example, SSL. Likewise, the correspondence channel is thought to be verified between Fog hubs and clients. To associate between the client and the mist hubs, the current conventions, for example, CoAP, are utilized which are viewed as the promising convention for IoT [20], notwithstanding verification of the clients at the mist hub.

D. Definition and Notation

To get to control, the information proprietor must allot significant credits to each document. The document's traits are equivalent to the one gathering's properties. To refresh the qualities, each property has a variant number, which will be demonstrated later. Fog servers have a duplicate of a gathering property history list (GAtH), as we will see later. The GAtH contains the characteristics' advancement history and the PRE keys utilized. A PRE-key permits the information proprietor to allot re-encryption activities to the Fog hub without uncovering the information substance. Also, one virtual trait, indicated by AttV, must be resolved for the key's administration. AttV is the essential quality in each datum record's properties and client's entrance structure, and won't be refreshed. The client has a completely mystery key, while the mist and cloud have a somewhat client's mystery key since they come up short on a mystery key part relating to a virtual trait, where that AttV is obscure for the mist and cloud. The objective of AttV is to empower the Fog to refresh the mystery key without uncovering it.

V. CONCLUSIONS AND FUTURE WORK

The present investigation intended to plan a safe information sharing system for a Fog domain. This system accomplished fine-grained get to control, information classification, client repudiation, and impact opposition. Our proposed structure

lays on a blend of KP-ABE and PRE-strategies. The commitment of the examination was the affirmation that our framework beat the cloud-based information sharing design. Our structure gives high adaptability and information partaking continuously and with low inactivity. The discoveries of this examination demonstrate that our framework beats cloud-based information offering frameworks to IoT quicker preparing time. The reproduction results additionally show that our framework reacts quicker to client demands than old style cloud frameworks.

VI. REFERENCES

[1]. Firdhous, M., Ghazali, O. and Hassan, S. (2014) Fog Computing: Will It Be the Future of Cloud Computing. Proceedings of the 3rd International Conference on Informatics & Applications, Kuala Terengganu, Malaysia, 8-15.

[2]. Bonomi, F., Milito, R., Zhu, J. and Addepalli, S. (2012) Fog Computing and IoT Role in the Internet of Things. Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, Helsinki, Finland, 17 August 2012, 13-16.

[3]. Stojmenovic, J. (2014) Fog Computing: A Cloud to the Ground Support for Smart Things and Machine-to-Machine Networks. Australasian Telecommunication Networks and Applications Conference (ATNAC), Southbank, VIC, 26-28 November 2014, 117-122. https://doi.org/10.1109/atnac.2014.7020884

[4]. Luan, T., Gao, L., Li, Z., Xiang, Y., We, G. and Sun, L. (2016) A View of Fog Computing from Networking Perspective. ArXivPrepr. ArXiv160201509.

[5]. Dastjerdi, A., Gupta, H., Calheiros, R., Ghosh, S. and Buyya, R. (2016) Fog Computing: Principals, Architectures, and Applications. ArXivPrepr. ArXiv160102752.

[6]. Yi, S., Hao, Z., Qin, Z., and Li, Q. (2015) Fog Computing: Platform and Applications. 2015 3rd IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb), Washington DC, 12-13 November 2015, 73-78. https://doi.org/10.1109/hotweb.2015.22

[7]. Scale, M. (2009) Cloud Computing and Collaboration. Library Hi Tech News, 26, 10-13. https://doi.org/10.1108/07419050911010741

[8]. Thilakanathan, D., Chen, S., Nepal, S. and Calvo, R. (2014) Secure Data Sharing in the Cloud. In: Nepal, S. and Pathan, M., Eds., Security, Privacy and Trust in Cloud Systems, Springer, Berlin, Heidelberg, 45-72.

[9]. Yu, S., Wang, C., Ren, K. and Lou, W. (2010) Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing. 2010 Proceedings IEEE INFOCOM, San Diego, CA, 14-19 March 2010, 1-9. https://doi.org/10.1109/infcom.2010.5462174

[10]. Wu, Y., Wei, Z. and Deng, R. (2013) Attribute-Based Access to Scalable Media in Cloud-Assisted Content Sharing Networks. IEEE Transactions on Multimedia, 15, 778-788. https://doi.org/10.1109/TMM.2013.2238910

[11]. Liu, Q., Wang, G. and Wu, J. (2014) Time-Based Proxy Re-Encryption Scheme for Secure Data Sharing in a Cloud Environment. Information Sciences, 258, 355-370. https://doi.org/10.1016/j.ins.2012.09.034

[12]. Tu, S., Niu, S., Li, H., Yun, X.-M. And Li, M. (2012) Fine-Grained Access Control and Revocation for Sharing Data on Clouds. 2012 IEEE 26th International Parallel and Distributed Processing Symposium Workshops & Ph.D. Forum (IPDPSW), Shanghai, 21-25 May 2012, 2146-2155. https://doi.org/10.1109/ipdpsw.2012.265

[13]. Yang, Y. and Zhang, Y. (2011) A Generic Scheme for Secure Data Sharing in Cloud. 2011 40th International Conference on Parallel Processing Workshops (ICPPW), Taipei City, 13-16 September 2011, 145-153. https://doi.org/10.1109/ICPPW.2011.51

[14]. Hur, J. (2013) Improving Security and Efficiency in Attribute-Based Data Sharing. IEEE Transactions on Knowledge and Data Engineering, 25, 2271-2282. https://doi.org/10.1109/TKDE.2011.78

[15]. Samanthula, B., Howser, G., Elmehdwi, Y. and Madria, S. (2012) An Efficient and Secure Data Sharing Framework Using Homomorphic Encryption in the Cloud. Proceedings of the 1st International Workshop on Cloud Intelligence, Istanbul, Turkey, 31 August 2012, Article No. 8. https://doi.org/10.1145/2347673.2347681

[16]. Zhang, R. and Chen, P. (2012) A Dynamic Cryptographic Access Control Scheme in Cloud Storage Services. 8th International Conference on Computing and Networking Technology (ICCNT), Gyeongju, 27-29 August 2012, 50-55.

[17]. Do, J., Song, Y. and Park, N. (2011) Attribute-Based Proxy Re-Encryption for Data Confidentiality in Cloud Computing Environments. 2011 First ACIS /JNU International Conference on Computers, Networks, Systems and Industrial Engineering (CNSI), Jeju Island, 23-25 May 2011, 248-251.

[18]. Qiao, Z., Liang, S., Davis, S. and Jiang, H. (2014) Survey of Attribute-Based Encryption. 2014 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel /Distributed Computing (SNPD), Las Vegas, NV, 30 June-2 July 2014, 1-6.

[19]. Ateniese, G., Fu, K., Green, M. and Hohenberger, S. (2006) Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage. ACM Transactions on Information and System Security, 9, 1-30.