A Proxy Re-Encryption Technique to fortify data Collection in IOT based on Blockchain

Mr.Kumar M[1], Ms.Seemitha Prasad S[2], Mr.Yuvraj M[3], Dr.Aruna MG[4], Mrs.Anupama PV[5], Dr.Malatesh SH[6]
[1,2,3]*Student, Department of CSE, M.S Engineering College, Bengaluru, India*
[4]*Associate Professor, Department of CSE, M.S Engineering College, Bengaluru, India*
[5]*Assistant Professor, Department of CSE, M.S Engineering College, Bengaluru, India*
[6]*Professor and Head, Department of CSE, M.S Engineering College, Bengaluru, India*

Abstract **- The evolution of the Internet of Things has seen data sharing as one of its most useful applications in cloud computing. As eye-catching as this technology has been, data security remains one of the obstacles it faces since the wrongful use of data leads to several damages. In this article, we propose a proxy re-encryption approach to secure data sharing in cloud environments. Data owners can outsource their encrypted data to the cloud using identity-based encryption, while proxy re-encryption construction will grant legitimate users access to the data. With the Internet of Things devices being resource-constrained, an edge device acts as a proxy server to handle intensive computations. Also, we make use of the features of information-centric networking to deliver cached content in the proxy effectively, thus improving the quality of service and making good use of the network bandwidth. Further, our system model is based on blockchain, a disruptive technology that enables decentralization in data sharing. It mitigates the bottlenecks in centralized systems and achieves fine-grained access control to data. The security analysis and evaluation of our scheme show the promise of our approach in ensuring data confidentiality, integrity, and security.**

I.INTRODUCTION

The Internet of Things (IoT) has emerged as a technology that has great significance to the world nowadays and its utilization has given rise to an expanded growth in network traffic volumes over the years. It is expected that a lot of devices will get connected in the years ahead. Data is a central notion to the IoT paradigm as the data collected serves several purposes in applications such as healthcare, vehicular networks, smart cities, industries, and manufacturing, among others. The sensors measure a host of parameters that are very useful for stakeholders involved. Consequently, as enticing as IoT seems to be, its advancement has introduced new challenges to security and privacy. IoT needs to be secured against attacks that hinder it from providing the required services, in addition to those that pose threats to the confidentiality, integrity, and privacy of data. A viable solution is to encrypt the data before outsourcing to the cloud servers. Attackers can only see the data in its encrypted form when traditional security measures fail. In data sharing, any information must be encrypted from the source and only decrypted by authorized users in order to preserve its protection. Conventional encryption techniques can be used, where the decryption key is shared among all the data users designated by the data owner. The use of symmetric encryption implies that the same key is shared between the data owner and users, or at least the participants agree on a key. This solution is very inefficient. Furthermore, the data owners do not know in advance who the intended data users are, and, therefore, the encrypted data needs to be decrypted and subsequently encrypted with a key known to both the data owner and the users. This decrypt-and-encrypt solution means the data owner has to be online all the time, which is practically not feasible. The problem becomes increasingly complex when there are multiple pieces of data and diverse data owners and users.

II.EXISTING SYSTEM

A viable solution is to encrypt the data before outsourcing to the cloud servers. Attackers can only see the data in its encrypted form when traditional security measures fail. In data sharing, any information must be encrypted from the source and only decrypted by authorized users in order to preserve

its protection. Conventional encryption techniques can be used, where the decryption key is shared among all the data users designated by the data owner. The use of symmetric encryption implies that the same key is shared between the data owner and users, or at least the participants agree on a key. This solution is very inefficient. Furthermore, the data owners do not know in advance who the intended data users are, and, therefore, the encrypted data needs to be decrypted and subsequently encrypted with a key known to both the data owner and the users. This decrypt-and-encrypt solution means the data owner has to be online all the time, which is practically not feasible. The problem becomes increasingly complex when there are multiple pieces of data and diverse data owners and users. The re-encryption keys were not only bound to the users' identities but also to a specific ciphertext. This implied that the data owner had to create a different re-encryption key for each pair of data user and shared file. In hierarchical PRE instead of an identity-based PRE. These two schemes tend to be inefficient when multiple and complex data pieces are considered.

III.PROPOSED SYSTEM

To overcome the limitations of the existing systems, we make use of the features of information-centric networking to deliver cached content in the proxy effectively, thus improving the quality of service and making good use of the network bandwidth. Further, our system model is based on blockchain, a disruptive technology that enables decentralization in data sharing. It mitigates the bottlenecks in centralized systems and achieves fine-grained access control to data. The emergence of the IoT has made data sharing one of its most prominent applications. To guarantee data confidentiality, integrity, and privacy, we propose a secure identity- based PRE-data-sharing scheme in a cloud computing environment. Secure data sharing is realized with IBPRE technique, which allows the data owners to store their encrypted data in the cloud and share them with legitimate users efficiently. Due to resource constraints, an edge device serves as the proxy to handle the intensive computations. The scheme also incorporates the features of ICN to proficiently deliver cached content, thereby improving the quality of service and making great use of the network bandwidth. Then, we present a blockchain-based system model that allows for flexible authorization on encrypted data. Fine grained access control is achieved, and it can help data owners achieve privacy preservation in an adequate way. The analysis and results of the proposed model show how efficient our scheme is, compared to existing schemes.

IV.SYSTEM ARCHITECTURE

The system architecture provides a holistic view of the system to be built. It depicts the structure and organization of software components, their properties and the connections between them. The architectural design process is concerned with establishing a basic structural framework for a system. It involves identifying the major components of the system and communications between these components. The system architecture shown in the Fig. 5.1 has four components CSP, Data owner(DO), Data User(DU) and Trusted Authority(Blockchain)Our system model introduces a blockchain-based PRE approach to data sharing The edge devices serve as proxy nodes and provide re-encryption services to the authorized user(s). When the data is cached at the edge of the network, the edge devices provide services to users with high availability and performance.

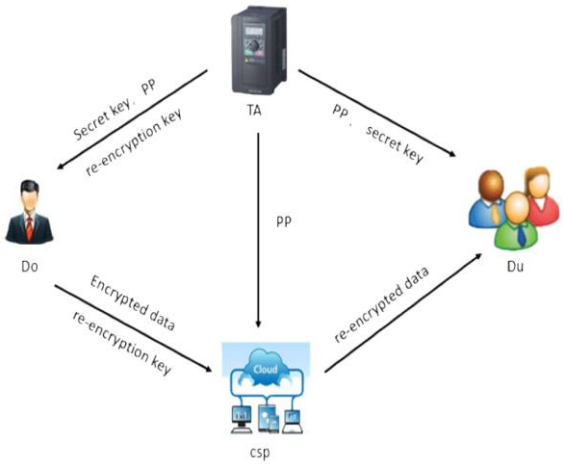

Fig 4.1 System Architecture

They receive the re-encryption key from the data owner, fetch the ciphertext from the CSP, and transform the ciphertext in the identity of the data user. It is an honest-but-curious entity. The blockchain serves as the trusted authority (TA) that initiates the system parameters. The TA also provides secret keys that are bound to the users' identities. By utilizing this distributed ledger, authenticity, transparency, and verifiability are achieved in the network, which

enhances the security and privacy of data. Data owners are therefore able to manage their data effectively. The blockchain network registers and issues membership keys to the data owner(s) and user(s). When a user requests data access, the owner generates a re-encryption key by using the identity of the user and sends it to the proxy server. Access rights and policies on the use of the data are instantiated and sent to the blockchain network. A data user is verified before access is granted.

A. SEQUENCE DIAGRAM

The sequence diagram is used primarily to show the interactions between objects in the sequential order that those interactions occur. Much like the class diagram, developers typically think the sequence diagrams are meant exclusively for them. However, an organization ‘s business staff can find sequence diagrams useful to communicate how the business currently works by showing how various business objects interact. Besides documenting an organizations current affair, a business-level sequence diagram can be used as a requirements document to communicate requirements for a future system implementation. The objectives involved in the operation are listed from left to right according to when they take part in the message sequence.

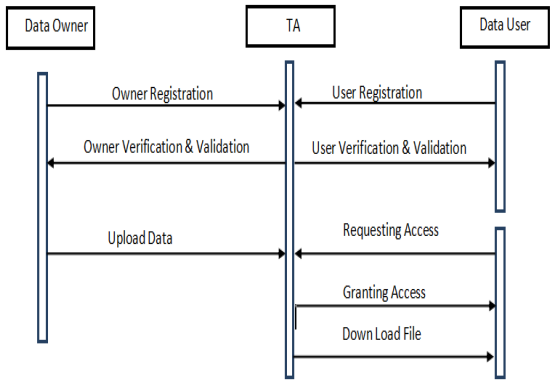

Fig 4.2 Sequence Diagram

V.METHODOLOGY

Our system model introduces a blockchain-based PRE approach to data sharing The edge devices serve as proxy nodes and provide re-encryption services to the authorized user(s). When the data is cached at the edge of the network, the edge devices provide services to users with high availability and performance. They receive the re-encryption key from the data owner, fetch the ciphertext from the CSP, and transform the ciphertext in the identity of the data user. It is an honest-but-curious entity. The blockchain serves as the trusted authority (TA) that initiates the system parameters. The TA also provides secret keys that are bound to the users’ identities. By utilizing this distributed ledger, authenticity, transparency, and verifiability are achieved in the network, which enhances the security and privacy of data. Data owners are therefore able to manage their data effectively. The blockchain network registers and issues membership keys to the data owner(s) and user(s). When a user requests data access, the owner generates a re-encryption key by using the identity of the user and sends it to the proxy server. Access rights and policies on the use of the data are instantiated and sent to the blockchain network. A data user is verified before access is granted.

SHA-256 ALGORITHM

SHA-2 (Secure Hash Algorithm 2), of which SHA-256 is a part, is one of the most popular hash algorithms around. A cryptographic hash, also often referred to as a “digest”, “fingerprint” or “signature”, is an almost perfectly unique string of characters that is generated from a separate piece of input text. For example, SHA-256 generates a 256-bit (32-byte) signature. Further down in this article, we will break down each step of SHA 256’s cryptographic algorithm and work through a real example by hand. Like we mentioned above, a cryptographic hash function generates a “fingerprint” for a given input string. For example, if we were to hash the entire text of JRR Tolkien’s “The Lord of The Rings” series using the SHA 256 algorithm, we would get a 256-bit output almost unique to that book’s text. If we changed even a single letter in the book, the output hash would be wildly different. It’s worth noting that we say the output of a hash is “almost unique” because there are a finite number of output strings. After all, the output of SHA-256 is always 256 bits long, which means it has a fixed size. The number of possible inputs, however, is infinite, meaning some inputs will hash to the same output. When this happens, it’s called a “collision”, and it is nearly impossible. After all, in SHA-256 there are 2^256 possible outputs. Let me write out that number for you:

115,792,089,237,316,195,423,570,985,008,687,907,853,269,984,665,640,564,039,4

The NSA, or National Security Agency, designed and published SHA-256 and the rest of the SHA-2 family of hash functions in 2001. You might be wondering: Because the United State Government helped create SHA-256, do they have some sort of "back-door" to break the encryption? The answer is "no". The algorithm is open source, so anyone can verify its security. While it's possible that there are exploitable vulnerabilities, no one has found them yet. At present, there isn't much you can do to SHA-256 apart from attempting a brute-force attack.

VI.PERFORMANCE EVALUATION

Performance analysis of an algorithm is done to understand how efficient the proposed system is compared to existing system that solves the same computational problem. Here we are comparing accuracy and time of existing

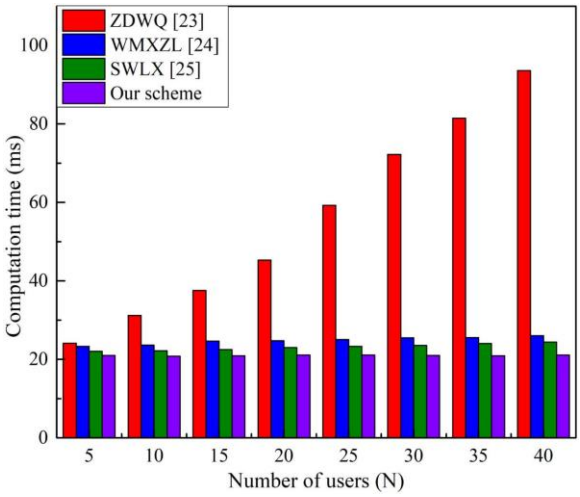

Fig 6.1 Performance Avaluation

REFERENCES

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," IEEE Commun. Surveys Tut., vol. 17, no. 4, pp. 2347–2376, Oct./Dec. 2015.

[2] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in Proc. Int. Conf. Theory Appl. Cryptographic Techn., Springer, May 1998, pp. 127–144.

[3] A. Shamir, "Identity-based cryptosystems and signature schemes," in Proc. Workshop Theory Appl. Cryptographic Techn., Springer, Aug. 1984, pp. 47–53.

[4] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. Int. Conf. Theory Appl. Cryptographic Techn., Springer, May 2004, pp. 506–522.

[5] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log," in NDSS, vol. 4. Citeseer, Feb. 2004,

[6] D. Balfanz et al., "Secret handshakes from pairing-based key agreements," in Proc. IEEE, Symp. Secur. Privacy, 2003, pp. 180–196.

[7] R. Canetti, S. Halevi, and J. Katz, "Chosen-ciphertext security from identity-based encryption," in Proc. Int. Conf. Theory Appl. Cryptographic Techn., Springer, 2004, pp. 207–222.

[8] T. Koponen *et al.*, "A data-oriented (and beyond) network architecture," in *Proc. Conf. Appl., Techn., Architectures, Protoc. Comput. Commun.*, Aug. 2007, pp. 181–192.

[9] N. Fotiou, P. Nikander, D. Trossen, and G. C. Polyzos, "Developing information networking further: From PSIRP to pursuit," in *Proc. Int. Conf. Broadband Commun., Netw. Syst.*, Springer, Oct. 2010, pp. 1–13.

[10] C. Dannewitz, J. Golic, B. Ohlman, and B. Ahlgren, "Secure naming for a network of information," in *Proc. INFOCOM IEEE Conf. Comput. Commun. Workshops*,2010, pp. 1–6.

[11] A. Carzaniga, M. J. Rutherford, and A. L. Wolf, "A routing scheme for content-based networking," in *Proc. IEEE INFOCOM 2004*, vol. 2, 2004, pp. 918–928.

[12] I. Psaras, W. K. Chai, and G. Pavlou, "Probabilistic in-network caching for information-centric networks," in *Proc. 2nd ed. ICN Workshop Inform.Centric Netw.*, Aug. 2012, pp. 55–60.

[13] Y. Sun *et al.*, "Trace-driven analysis of ICN caching algorithms on video on-demand workloads," in *Proc.10thACMInt.Conf. Emerging Netw.Exp. Technol.*, Dec. 2014, pp. 363–376.