International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 03 | March - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Decentralized Secure Data Sharing for Mobile Edge Computing Using Blockchain

A. Amulya[1], K. Uday Kiran[2], K. Nagesh[3]
[1]Assistant Professor, Mahatma Gandhi Institute of Technology
[2,3]UG Student, Mahatma Gandhi Institute of Technology

Abstract: **Mobile Edge Computing (MEC) is a promising technology that provides high bandwidth and low latency for mobile users. However, it raises significant security challenges, particularly in ensuring the secure transfer of sensitive user data across subnetworks. This paper presents a blockchain-based key management scheme for MEC that enhances security and trust in dynamic mobile environments. The proposed approach allows mobile devices to communicate securely by encrypting data with public keys stored on the blockchain, ensuring efficient key management and mitigating potential attacks. Theoretical analysis demonstrates improved security, while experimental results validate the scheme's efficiency and scalability.**

Index Terms: Blockchain, Mobile Edge Computing, Key Management, Cybersecurity, Secure Communication

I. INTRODUCTION

With the growing reliance on mobile cloud services, Mobile Edge Computing (MEC) has emerged as a crucial technology to enable low-latency applications. MEC extends cloud computing to the network edge, closer to mobile users, ensuring faster processing and better user experience. However, security concerns arise when transferring sensitive data among untrusted nodes.

A major challenge in MEC is ensuring group communication and key transfer when devices move between subnetworks, leading to additional computation, communication, and storage overheads.

Group key management schemes in MEC must manage both dynamic group membership (members joining or leaving) and dynamic member location (movement between network areas).

Key challenges include managing large numbers of mobile devices, ensuring real-time performance during the rekeying process, and addressing security concerns in centralized key management systems.

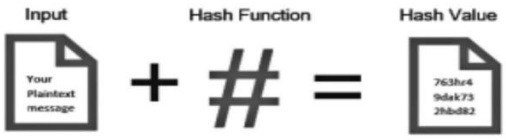
A. Problem Statement

In this project we have a two different users. Mobile user1 have a upload a data and it will have a how to transfer keys to members moving from one sub network (blocks) to another while still remaining in communication. Mobile user1 and Mobile user2 data has sharing.

This paper aims to improve the security of key management in MEC network. In order to deal with all the challenges mentioned above, we analyze the characteristics of key management schemes and block chain technique, and then we construct a block chain has key management scheme. In the proposed scheme, we first partition blocks (i.e., subnetworks), according to the number of members (i.e., users). Then in each group, all members maintain a blockchain network.

B. EXISTING SYSTEM

SHA-256 ALGORITHM (SECURE HASHING ALGORITHM) SHA-256 algorithm is a hashing algorithm which perform on data in one-way and it is developed by Ron Rivest. It is an evolution of previous algorithms such as SHA 0, SHA 1, SHA 256, SHA 384. Hashing is also known as compression or message summary function which takes the entire variable length and change it into a binary sequence of fixed length. This hash function is designed in such way that it is impossible to reverse the process, hence it is called one direction. The concept of hashing algorithm is shown in Figure - Increased storage requirements, leading to performance issues.



One major challenge for MEC is managing devices that frequently join, leave, or move between subnetworks. As mobility increases, the system requires a dynamic, lightweight rekeying mechanism to ensure backward and forward secrecy without introducing significant computational overhead.

The mobility of devices within a MEC environment increases communication overhead due to frequent key updates. This overhead can severely impact system performance, especially for real-time services like video streaming or virtual reality applications.

MEC infrastructures are expected to serve large-scale networks with potentially thousands of mobile devices. This increases the complexity of managing group keys across the entire system.

- High Computation Overhead
- Vulnerability to Attacks
- Limited Flexibility for Group Management
- Increased Storage Requirements

II. PROPOSED SYSTEM

A. Architecture of Proposed System

In distributed key management schemes, there are no explicit key distribution center (KDC) and all the members can devote to the management. The distributed schemes can help to unify the workload of key management and reduce the requirement of central entities. proposed a blockchain-based key management scheme in named data network to solve the problem of lacking mutual trust between sites without trust users. It has an efficient key management scheme for block chain. With the help of group-based keys within the context of clustered and distributed key management framework.

Implement a hierarchical key distribution model where a centralized MEC controller handles key updates at higher levels, while edge nodes manage local group keys for nearby devices. This decentralized structure helps reduce communication overhead by localizing key management, minimizing the need for global updates when a device moves across subnetworks.

B. Advantages of Proposed System

Use a hybrid cryptosystem that combines symmetric encryption for fast operations with asymmetric encryption for secure key exchange. This approach can reduce computation time and provide a balance between performance and security. This system should enable efficient key generation, distribution, and storage even in highly mobile environments.

Implement a hierarchical key distribution model where a centralized MEC controller handles key updates at higher levels, while edge nodes manage local group keys for nearby devices. This decentralized structure helps reduce communication overhead by localizing key management, minimizing the need for global updates when a device moves across subnetworks.

- Reducing storage cost.
- Providing more security.

C. System Requirement Specifications

- The front-end layer of the application is built using J2EE technologies, specifically JSP and Servlets, to deliver dynamic web content and seamless user interaction.
- JSP is used for creating interactive and responsive user interfaces, while Servlets handle back-end logic and facilitate communication between the client and server.
- The back-end is powered by MySQL 5.5, ensuring efficient data storage, retrieval, and management for the application.
- MySQL 5.5 provides a robust and scalable database solution to handle the application's data operations and queries seamlessly.

D. Hardware Requirements Specifications

- Processor
- Memory (2GB DDRAM)
- Hard Disk (250GB)

III. LITERATURE SURVEY

Numerous research papers have explored secure data sharing using blockchain. Below are some key studies related to our work:

In the paper Secure and Efficient Data Sharing on Blockchain by authors Zhang et al. (2021), the authors propose an encryption mechanism that leverages smart contracts to ensure data integrity and privacy in blockchain networks. The study highlights how blockchain can provide secure data sharing while maintaining transparency. However, the authors note that the approach comes with high computational overhead due to the complexity of encryption processes, which may limit its efficiency in resource constrained environments [1].

In the paper Blockchain-Based Data Sharing for IoT Devices by authors Kumar et al. (2021), the authors design a lightweight framework for IoT systems using blockchain technology to securely share data between users. The proposed solution aims to address security challenges in IoT environments while ensuring efficient data management. Despite its benefits, the authors emphasize the challenge of limited scalability when applied to large-scale IoT networks, which could affect its real-world adoption [2].

In the paper Blockchain for Privacy-Preserving Database Sharing by authors Singh and Verma (2022), the authors focus on privacy-enhancing techniques using Zero Knowledge Proofs (ZKP) to ensure secure data sharing. The study highlights the effectiveness of ZKP in enhancing data privacy. However, the authors note that ZKP protocols are complex, making their implementation time-consuming [3].

In the paper Blockchain-Based Secure Multi-Party Data Sharing by authors Park et al. (2022), the authors develop a consensus algorithm to enable efficient data sharing between multiple users in a blockchain environment. The research addresses the challenge of securely managing multi-party transactions. Despite its benefits, the approach remains vulnerable to Sybil attacks under certain network conditions [4].

In the paper Privacy-Preserving Blockchain-Based Database Framework by authors Chen and Liu (2023), the authors emphasize privacy-preserving data transactions by incorporating differential privacy and blockchain technology. The proposed framework ensures data confidentiality during real-time data exchange. However, the authors highlight the challenge of high overhead due to privacy mechanisms, which can affect real-time processing [5].

In the paper Secure Data Exchange Using Blockchain in Distributed Systems by authors Wu et al. (2023), the authors develop a model integrating homomorphic encryption to facilitate secure data exchange over distributed systems. This approach ensures end-to-end data security in decentralized networks. Nevertheless, the authors identify that the computational cost of homomorphic encryption limits the model's scalability [6].

IV. SYSTEM DESIGN AND METHODOLOGY

The proposed system consists of multiple components, including mobile users, an admin, a cloud server, and a blockchain ledger. These components collaborate to manage key generation, distribution, and verification. The system follows a structured methodology to ensure data security.

A. Key Modules

1. User Interface Design - Secure authentication and access control.

2. Admin Module - Maintains key logs and user details.

3. Cloud Server - Stores encrypted data and verifies key integrity.

4. Mobile Users - Generate and share public-private key pairs securely.

5. Blockchain Network - Ensures transparency and security in key exchanges.

B. Distributed Key Management (DKM)

Long Short-Term Memory (LSTM) networks are a specialized form of Recurrent Neural Networks (RNNs) designed to address the limitations of traditional RNNs in capturing long-term dependencies. Introduced by Hochreiter and Schmidhuber in 1997, LSTMs overcome issues like vanishing and exploding gradients, making them highly effective for processing and predicting data with complex temporal or sequential dependencies, such as text, time-series data, and speech.

C. Blockchain-based Key Management

Blockchain technology provides a promising solution to distributed key management, offering features such as decentralization, immutability, and transparency. By using blockchain, key management becomes more secure, efficient, and scalable, particularly in decentralized networks like Named Data Networks (NDN), where mutual trust between nodes or users is often lacking.

V. UML DIAGRAMS

The following UML diagrams illustrate the system's architecture:

- Use Case Diagram - Shows interactions between users, admin, and cloud server.

-Class Diagram - Represents relationships between key management entities.

- Activity Diagram - Details the step-by-step process of secure data sharing.

- Sequence Diagram - Depicts the sequence of operations in key management.

- Deployment Diagram - Outlines the deployment of the proposed system.

VI. RESULTS AND CONCLUSION

This study demonstrates that a blockchain-based key management system can enhance security, efficiency, and scalability in MEC networks. Our approach reduces computational costs while ensuring secure, trustless key exchanges between mobile users.

A. Future Scope

- Integration with AI models for anomaly detection.

- Optimization for large-scale networks to improve processing efficiency.

- Enhancement with Zero-Knowledge Proofs (ZKP) for privacy-preserving transactions.

VII. ACKNOWLEDGMENT

We express our sincere gratitude to Prof. G. Chandramohan Reddy (Principal, MGIT) and Dr. D. Vijaya Lakshmi (HOD, IT Dept., MGIT) for their invaluable support. We also extend our thanks to our project guide Mrs. A. Amulya and our Project Coordinator Dr. N. Sree Divya for their guidance throughout the research.

IX. REFERENCES

[1] R. Roman, J. Lopez, “Security Challenges in Mobile Edge Computing,” Future Generation Computer Systems, 2018.

[2] S. Yi, Z. Qin, Q. Li, “Fog Computing Security and Privacy Issues,” International Wireless Algorithms Conference, 2015.

[3] C. Esposito, M. Ficco, “Distributed Key Management for Event Notification in MEC,” IEEE Transactions on Secure Computing, 2020.

[4] X. Li, P. Jiang, “Blockchain Security for Mobile Networks,” Future Generation Computer Systems, 2017.