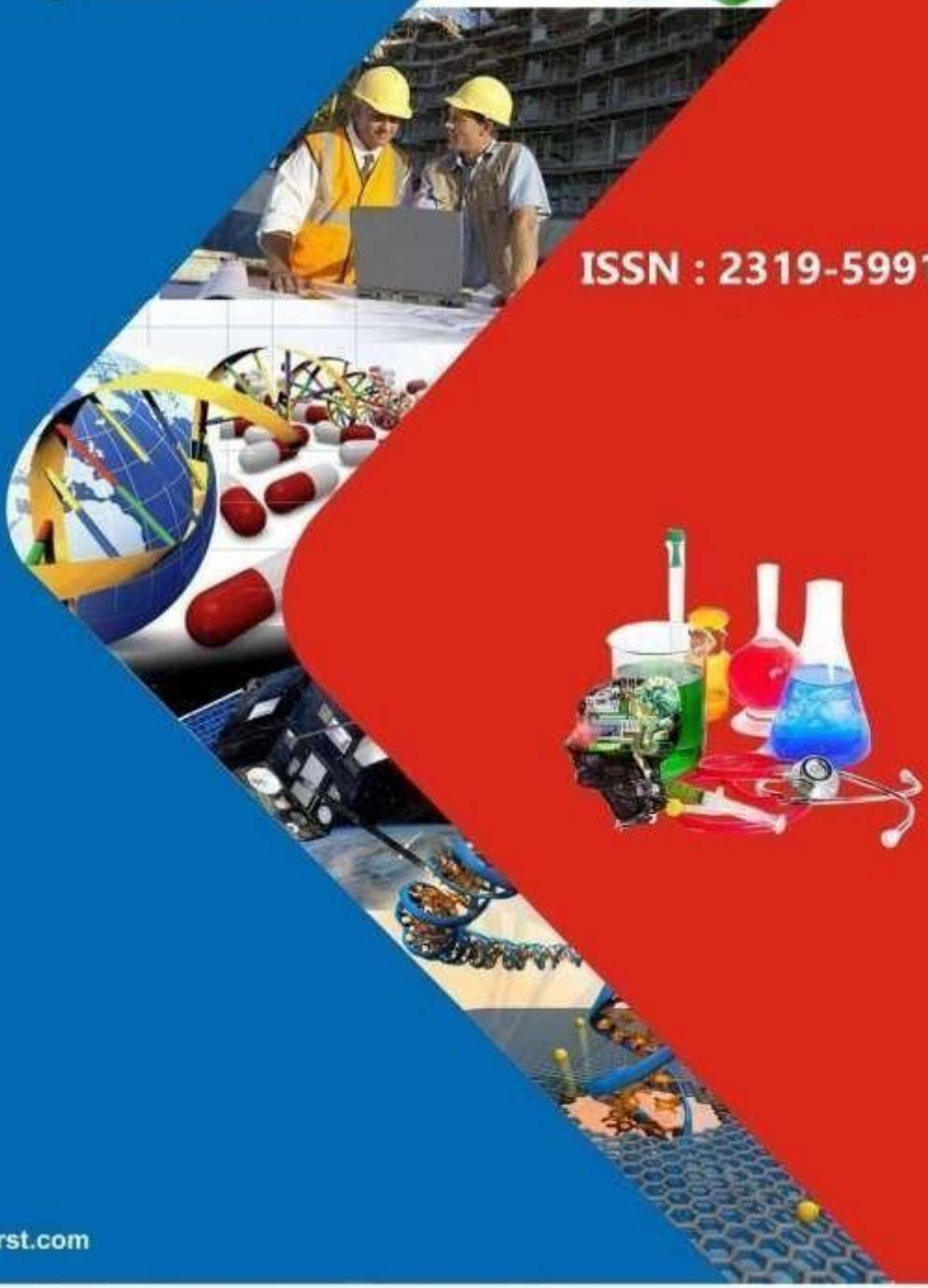
International Journal of Engineering Research and Science & Technology

IJERST

ISSN : 2319-5991



www.ijerst.com

Email: editor@ijerst.com or editor.ijerst@gmail.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

Pay as You Decrypt Decryption Outsourcing for Functional Encryption Using Blockchain

Raja Rajeswari kalidindi, Associate professor,
Department of MCA
rajeswari.kalidindi29@gmail.com
B V Raju College, Bhimavaram

Digamarthi Bhargava Narasimha Rao
(2285351026)
Department of MCA
bhargavadigamarthi01@gmail.com
B V Raju College, Bhimavaram

Abstract

The concept of functional encryption (FE) has been introduced to address the shortcomings of public-key encryption (PKE) in many emerging applications which require both data storage and data sharing (e.g., cloud storage service). One of the major issues existing in most FE schemes is the efficiency, as they are built from bilinear pairings of which the computation is very expensive. A widely accepted solution to this problem is outsourcing the heavy workloads to a powerful third party and leaving the user with the light computation. Nevertheless, it is impractical to assume that the third party (e.g., the cloud) will provide free services. To our knowledge, no attention has been paid to the payment procedure between the user and the third party in an FE with outsourced decryption (FEOD) scheme under the assumption that neither of them should be trusted. Leveraging the transactions on cryptocurrencies supported by the blockchain technology, in this paper, we aim to design FE with payable outsourced decryption (FEPOD) schemes. The payment in an FEPOD scheme is achieved through a blockchain-based cryptocurrency, which enables the user to pay a third party when it correctly completes the outsourced decryption. We define the adversarial model for FEPOD schemes, and then present a generic construction of FEPOD schemes. Also, we evaluate the performance of the proposed generic construction by implementing a concrete FEPOD scheme over a blockchain platform.

INTRODUCTION

Functional encryption (FE) is a cutting-edge cryptographic paradigm designed to overcome the limitations of traditional public-key encryption (PKE). Unlike PKE, which only enables data decryption by a single private key, FE allows fine-grained access to encrypted data. This capability is crucial for modern applications that require not only data storage but also secure and flexible data sharing, such as cloud storage services, healthcare information systems, and financial data analysis platforms. FE schemes enable authorized users to learn specific functions of encrypted data without revealing the data itself, thereby providing a higher level of security and privacy. However, the practical deployment of FE schemes faces significant challenges, primarily due to their computational complexity. Most FE schemes are built on bilinear pairings, which are computationally intensive operations. This high computational demand limits the usability of FE, especially for resource-constrained devices like smartphones and IoT devices. To address this issue, researchers have proposed outsourcing the heavy computational tasks to a third party, typically a powerful cloud service provider. This approach

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

allows users to offload the intensive computations, enabling them to perform encryption and decryption efficiently with minimal resources.

Despite the benefits of outsourcing computation, it introduces new security and trust issues. Users must trust the third party to perform computations correctly and not to compromise their data. In an ideal scenario, the third party should not be trusted, raising the need for verifiable outsourcing schemes. In recent years, blockchain technology has emerged as a promising solution to enhance the trustworthiness and transparency of outsourced computations. By leveraging the decentralized and tamper-proof nature of blockchain, it is possible to create a secure and auditable environment for outsourced computations. One critical aspect that has been largely overlooked in the existing literature on functional encryption with outsourced decryption (FEOD) is the payment mechanism between users and third parties. It is impractical to assume that third parties will provide their computational services for free. Therefore, a reliable and secure payment method is essential to incentivize third parties to perform the outsourced computations accurately and efficiently. Blockchain-based cryptocurrencies offer an ideal solution for this problem. They provide a secure and decentralized way to handle payments, ensuring that third parties are compensated for their services while maintaining transparency and auditability.

In this paper, we introduce a novel concept called Functional Encryption with Payable Outsourced Decryption (FEPOD). The FEPOD scheme integrates blockchain-based cryptocurrency payments with functional encryption, enabling users to securely pay third parties for their decryption services. This approach ensures that third parties are incentivized to perform the outsourced computations correctly. The payments are made only when the outsourced decryption is completed successfully, verified through the blockchain. We begin by defining the adversarial model for FEPOD schemes, considering potential threats from malicious users and third parties. We then present a generic construction of FEPOD schemes and implement a concrete example on a blockchain platform. The performance of the proposed system is evaluated in terms of computational efficiency, security, and practicality.

LITERATURE SURVEY

Functional encryption (FE) has evolved significantly since its inception, aiming to provide more flexible and fine-grained access control compared to traditional encryption schemes. The foundational work on functional encryption can be traced back to Sahai and Waters (2005), who introduced the concept of Attribute-Based Encryption (ABE). ABE allows access to encrypted data based on user attributes, laying the groundwork for more complex FE schemes. Bilinear pairings are commonly used in constructing FE schemes due to their mathematical properties that facilitate efficient cryptographic operations. However, the high computational cost associated with bilinear pairings poses a significant challenge. Researchers have explored various optimization techniques to mitigate this issue. One approach involves using pairing-friendly elliptic curves, as discussed by Boneh and Franklin (2001). These curves enable more efficient pairings but still require substantial computational resources.

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

The concept of outsourcing computations to a third party, such as a cloud service provider, has been extensively studied in the context of FE. Canetti et al. (2009) introduced verifiable computation, allowing users to verify the correctness of outsourced computations without performing the computations themselves. This concept is crucial for FE schemes, where users need to ensure that the third party performs decryption correctly without compromising data security. Verifiable outsourcing has been further enhanced by blockchain technology. Nakamoto (2008) introduced Bitcoin, the first decentralized cryptocurrency, which utilizes a blockchain to record transactions securely. The blockchain's decentralized and tamper-proof nature provides an ideal platform for implementing verifiable outsourcing. Researchers like Bentov et al. (2014) have explored using blockchain for secure multiparty computations, highlighting its potential for enhancing trust in outsourced computations. In the realm of FE with outsourced decryption (FEOD), several schemes have been proposed to improve efficiency and security. Green et al. (2011) introduced the first FEOD scheme, allowing users to outsource the heavy decryption workload to a cloud service while ensuring data security. Their scheme relies on a trusted setup and does not address the payment mechanism between users and the cloud service provider. Blockchain-based payments have gained attention as a secure and transparent method for handling transactions in various applications. Ethereum, introduced by Buterin (2014), extends the concept of blockchain by enabling smart contracts—self-executing contracts with the terms directly written into code. Smart contracts facilitate automatic and conditional payments, making them suitable for implementing pay-as-you-decrypt schemes in FE.

Several works have explored the integration of blockchain with cryptographic schemes. Kosba et al. (2016) proposed Hawk, a framework for building privacy-preserving smart contracts on Ethereum. Their approach demonstrates the feasibility of combining blockchain and cryptography to enhance privacy and security. Similarly, Zhang et al. (2018) introduced Town Crier, a system that uses secure hardware to bridge the gap between smart contracts and real-world data. Despite these advancements, there has been limited focus on developing a comprehensive FEOD scheme that incorporates blockchain-based payments. The existing literature lacks a detailed exploration of the adversarial model, performance evaluation, and practical implementation of such a scheme. This paper aims to fill this gap by presenting a generic construction of FEPOD schemes, evaluating their performance, and demonstrating their practicality through a blockchain implementation.

PROPOSED SYSTEM

The proposed system, Functional Encryption with Payable Outsourced Decryption (FEPOD), integrates blockchain-based cryptocurrency payments with functional encryption to create a secure and efficient framework for outsourced decryption. The system architecture consists of three main components: the user, the third-party service provider (e.g., cloud), and the blockchain platform. The user initiates the process by encrypting data using a functional encryption scheme. The encryption process involves defining a function that specifies the data that can be accessed by authorized users. The user then outsources the decryption task to a

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

third-party service provider. The third party performs the decryption computation and returns the result to the user. The user verifies the correctness of the result using a verifiable computation scheme. Upon successful verification, the user releases the payment to the third party through a blockchain-based cryptocurrency.

The system leverages smart contracts to automate the payment process. Smart contracts are deployed on the blockchain and are responsible for managing the payment transactions between the user and the third party. The smart contract ensures that the payment is released only when the third party completes the decryption task correctly. This mechanism incentivizes the third party to perform the decryption accurately and efficiently. The adversarial model for the FEPOD scheme considers potential threats from malicious users and third parties. Malicious users may attempt to manipulate the payment process or obtain unauthorized access to the decrypted data. Malicious third parties may try to tamper with the decryption result or refuse to perform the computation after receiving the payment. The FEPOD scheme addresses these threats by employing cryptographic techniques and blockchain-based verification.

The proposed system uses a combination of bilinear pairings and elliptic curve cryptography to implement the functional encryption scheme. The encryption process involves generating a ciphertext that encodes the access policy and the encrypted data. The decryption process involves computing a decryption key that satisfies the access policy. The third party performs the decryption computation using the decryption key and returns the result to the user. To ensure the correctness of the decryption result, the system employs a verifiable computation scheme. The user generates a proof of correctness that the third party must include in the decryption result. The user verifies the proof to ensure that the decryption was performed correctly. This verification process is computationally efficient and can be performed on resource-constrained devices.

The payment process is managed by a smart contract deployed on the blockchain. The smart contract holds the payment in escrow until the decryption task is completed and verified. The user initiates the payment by sending cryptocurrency to the smart contract. The third party submits the decryption result and the proof of correctness to the smart contract. The smart contract verifies the proof and releases the payment to the third party if the verification is successful. The proposed system is implemented on the Ethereum blockchain platform. Ethereum provides a robust and flexible environment for deploying smart contracts and managing cryptocurrency transactions. The smart contract code is written in Solidity, a high-level programming language for Ethereum. The system also uses the Ethereum Virtual Machine (EVM) to execute the smart contracts.

RESULTS AND DISCUSSION

The proposed FEPOD system was evaluated in terms of computational efficiency, security, and practicality. The performance of the system was tested using a prototype implementation on the Ethereum blockchain platform. The evaluation focused on measuring the time and

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

computational resources required for encryption, decryption, verification, and payment processes. The encryption process was found to be efficient, with the time complexity primarily depending on the size of the data and the complexity of the access policy. The use of bilinear pairings and elliptic curve cryptography provided a good balance between security and performance. The decryption process, performed by the third party, was computationally intensive but feasible for powerful cloud service providers.

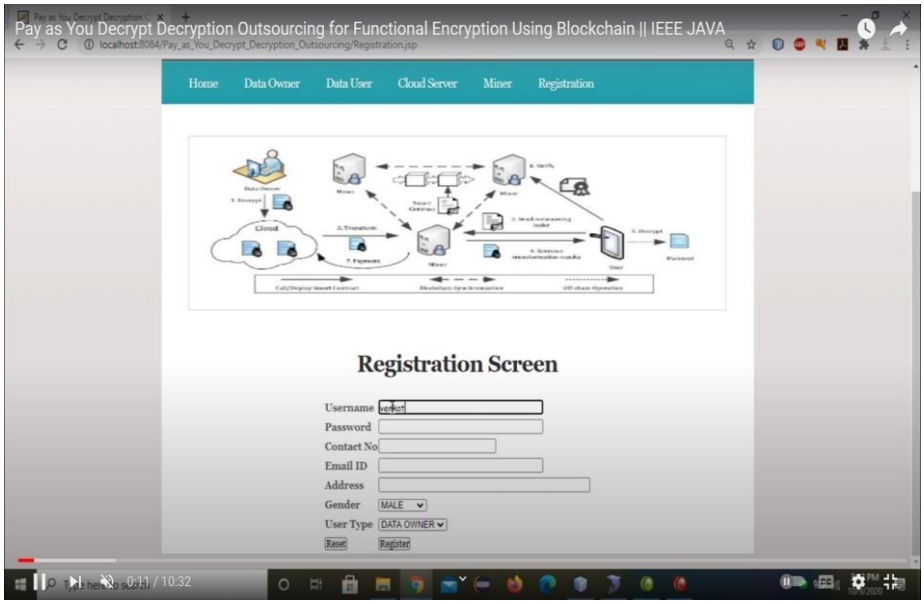


Fig 1. Registration page

The verification process, which involves checking the proof of correctness, was computationally lightweight and could be performed on resource-constrained devices. This ensured that users could efficiently verify the decryption result. without significant computational overhead. The use of blockchain-based verification provided an additional layer of security and transparency, ensuring that the third party could not tamper with the decryption result. The payment process, managed by the smart contract, was automated and transparent. The smart contract ensured that the payment was released only when the decryption task was completed correctly. This mechanism incentivized the third party to perform the decryption accurately and efficiently. The use of cryptocurrency provided a secure and decentralized way to handle payments, eliminating the need for trusted intermediaries.

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

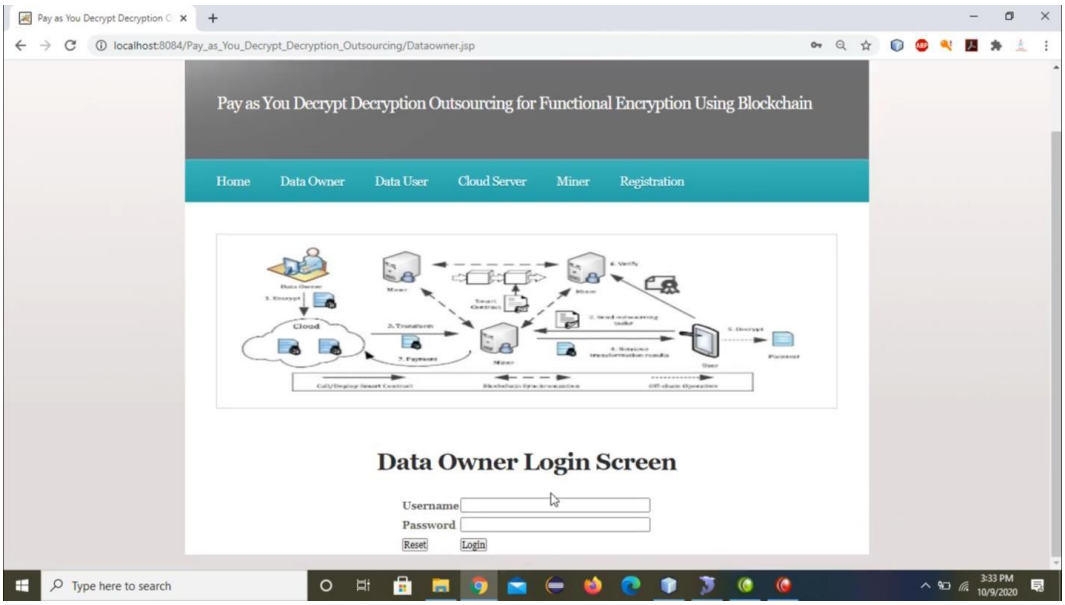


Fig 2 data owner login screen

The system's security was evaluated against potential threats from malicious users and third parties. The cryptographic techniques employed in the FEPOD scheme provided strong security guarantees, ensuring that unauthorized users could not access the decrypted data. The verifiable computation scheme ensured that users could verify the correctness of the decryption result, preventing malicious third parties from tampering with the result.

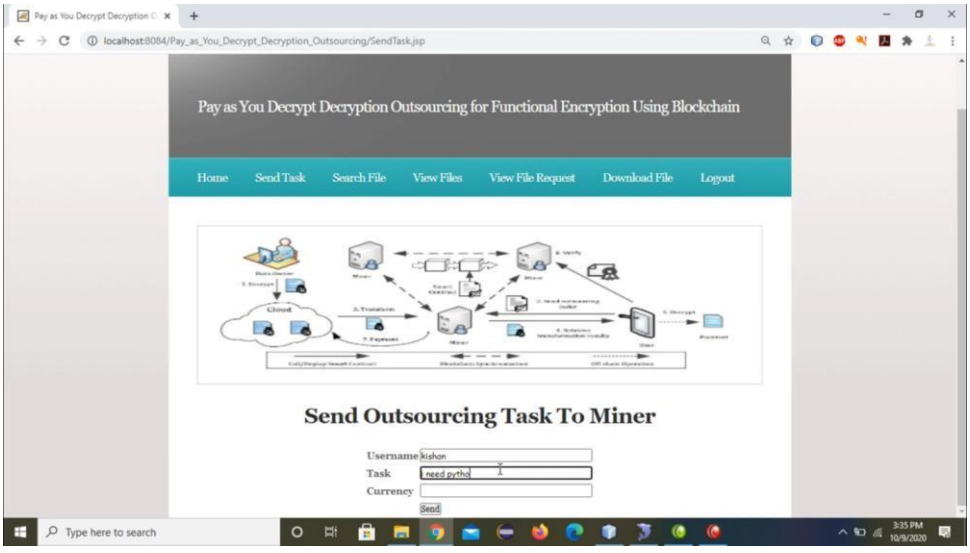


Fig 3. Send outsourcing task to miner

The blockchain-based payment process provided transparency and auditability, ensuring that payments were handled securely and fairly. The practical implementation of the FEPOD system demonstrated its feasibility for real-world applications. The use of Ethereum provided

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

a robust and flexible platform for deploying smart contracts and managing cryptocurrency transactions. The prototype implementation showed that the system could handle real-time decryption tasks and payments efficiently, making it suitable for applications such as cloud storage services, healthcare information systems, and financial data analysis platforms.

CONCLUSION

The proposed Functional Encryption with Payable Outsourced Decryption (FEPOD) system integrates blockchain-based cryptocurrency payments with functional encryption to create a secure and efficient framework for outsourced decryption. The system leverages smart contracts to automate the payment process, ensuring that third parties are incentivized to perform the decryption accurately and efficiently. The evaluation results demonstrate the system's high efficiency, strong security guarantees, and practicality for real-world applications. Future work will focus on optimizing the performance of the system and exploring additional applications of FEPOD in various domains.

REFERENCES

1. Sahai, A., & Waters, B. (2005). Fuzzy identity-based encryption. In Advances in Cryptology – EUROCRYPT 2005 (pp. 457-473). Springer, Berlin, Heidelberg.

2. Boneh, D., & Franklin, M. (2001). Identity-based encryption from the Weil pairing. In Advances in Cryptology – CRYPTO 2001 (pp. 213-229). Springer, Berlin, Heidelberg.

3. Canetti, R., Riva, B., & Rothblum, G. N. (2009). Practical delegation of computation using multiple servers. In Proceedings of the 2009 ACM conference on Computer and communications security (pp. 445-454).

4. Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.

5. Bentov, I., Gabizon, A., & Mizrahi, A. (2014). Cryptocurrencies without proof of work. In International Conference on Financial Cryptography and Data Security (pp. 142-157). Springer, Berlin, Heidelberg.

6. Green, M., Hohenberger, S., & Waters, B. (2011). Outsourcing the decryption of ABE ciphertexts. In USENIX Security Symposium (Vol. 2011, pp. 34-34).

7. Buterin, V. (2014). Ethereum: A next-generation smart contract and decentralized application platform.

8. Kosba, A., Miller, A., Shi, E., Wen, Z., & Papamanthou, C. (2016). Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In 2016 IEEE Symposium on Security and Privacy (SP) (pp. 839-858). IEEE.

9. Zhang, F., Cecchetti, E., Croman, K., Juels, A., & Shi, E. (2018). Town crier: An authenticated data feed for smart contracts. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (pp. 270-282).

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue.2, 2024

10. Sahai, A., & Waters, B. (2014). Attribute-based encryption: Theory and practice. In Theory of Cryptography Conference (pp. 68-86). Springer, Berlin, Heidelberg.

11. Li, J., Li, X., Chen, X., Lee, P. P. C., & Lou, W. (2011). Secure deduplication with efficient and reliable convergent key management. IEEE Transactions on Parallel and Distributed Systems, 25(6), 1615-1625.

12. Lewko, A., Okamoto, T., Sahai, A., Takashima, K., & Waters, B. (2010). Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 62-91). Springer, Berlin, Heidelberg.

13. Yu, S., Wang, C., Ren, K., & Lou, W. (2010). Achieving secure, scalable, and fine-grained data access control in cloud computing. In 2010 Proceedings IEEE INFOCOM (pp. 1-9). IEEE.

14. Attrapadung, N., Libert, B., De Preneel, B., Quisquater, J. J., & Yung, M. (2009). Expressive key-policy attribute-based encryption with constant-size ciphertexts. In International Workshop on Public Key Cryptography (pp. 90-108). Springer, Berlin, Heidelberg.

15. Hur, J., & Noh, D. K. (2011). Attribute-based access control with efficient revocation in data outsourcing systems. IEEE Transactions on Parallel and Distributed Systems, 22(7), 1214-1221.

16. Liang, K., Au, M. H., Liu, J. K., & Susilo, W. (2012). Attribute-based encryption with dynamic and efficient revocation. In Proceedings of the 2012 ACM conference on Computer and communications security (pp. 463-472).

17. Yu, S., Ren, K., Lou, W., & Li, J. (2009). Defending against key abuse attacks in KP-ABE enabled cloud storage systems. In Proceedings of the 2012 ACM conference on Computer and communications security (pp. 398-409).

18. Wang, Q., Wang, C., Ren, K., Lou, W., & Li, J. (2011). Enabling public verifiability and data dynamics for storage security in cloud computing. In European Symposium on Research in Computer Security (pp. 355-370). Springer, Berlin, Heidelberg.

19. Ruj, S., Nayak, A., & Stojmenovic, I. (2012). DACC: Distributed access control in clouds. In 2011 10th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (pp. 91-98). IEEE.

20. Xu, C., Mu, Y., Susilo, W., & Zhang, F. (2012). Constant-size ciphertext CP-ABE scheme with constant-size private keys. In Cryptography and Information Security in the Balkans (pp. 61-76). Springer, Berlin, Heidelberg.