Block Chain Based Identity and Access Management: A New Paradigm for Cybersecurity

Ranga Premsai

Maryland, USA
Premsairanga809@gmail.com

Abstract

The rapid growth of transaction operation data has introduced significant challenges related to balancing the need for data sharing and ensuring privacy protection. As transactions increasingly become more complex and data-driven, the sharing of this data across various stakeholders, while maintaining confidentiality and integrity, has become a critical concern. The contradiction between the transparent and immutable nature of transaction data in distributed systems and the need for stringent privacy measures presents a unique problem for industries dealing with sensitive transaction information, such as finance, healthcare, and e-commerce.In response to this problem, a novel blockchain-based transaction operation data sharing scheme, named BBTDSS, is proposed for the Identity and access management system. This scheme is designed to leverage blockchain technology for secure data sharing while implementing a crop quantum homomorphic encryption algorithm to ensure that transaction data remains private during processing and sharing. Blockchain's decentralized and immutable characteristics provide a robust framework for transparent data management, enabling multiple participants to access shared data without the need for a trusted central authority. However, the challenge of protecting sensitive transaction information is addressed by integrating homomorphic encryption, which allows computations to be performed on encrypted data without decrypting it. This ensures that the privacy of individual transaction details is maintained, even while enabling the secure sharing and processing of data across different parties.The introduction of crop quantum homomorphic encryption further enhances the security and scalability of the system by harnessing the power of quantum computing principles to perform more efficient and secure encryption operations. This approach significantly reduces the computational burden typically associated with traditional encryption methods, making the system more efficient and practical for real-time transaction processing.In essence, the BBTDSS scheme provides a solution to the ongoing challenge of data privacy and secure sharing by combining the strengths of blockchain and advanced encryption techniques. It ensures that transaction operation data can be shared securely among authorized parties, while safeguarding sensitive information from unauthorized access, thus paving the way for more transparent, efficient, and privacy-preserving transaction systems in various domains.

Keywords: Blockchain-Based Transaction Operation Data Sharing Scheme, Crop Quantum Homomorphic Encryption Algorithm, Identity And Access Management

I. INTRODUCTION

The rapid proliferation of transaction operation data in today’s digital economy has led to a significant challenge: how to effectively balance the need for secure data sharing with the protection of sensitive information. As transactions become more complex and data-driven, the demand for sharing transaction data

across multiple stakeholders—including service providers, customers, regulators, and financial institutions—has increased. However, this sharing must occur within a framework that ensures the confidentiality, integrity, and privacy of the data. In many industries, such as finance, healthcare, and e-commerce, the sensitive nature of transaction data adds a layer of complexity to the problem, as stakeholders must adhere to strict privacy regulations while ensuring that data is accessible when needed.One of the most promising solutions to address these issues is blockchain technology. Blockchain offers a decentralized and immutable ledger that ensures the transparency and integrity of data, eliminating the need for a trusted central authority. While blockchain's transparency makes it ideal for securely managing and sharing transaction data, it also presents a unique challenge—how to maintain the privacy of sensitive information. The transparent nature of blockchain means that all transaction details are visible to participants, which may expose private or confidential information if not properly secured.To solve this issue, we propose a novel solution: a **Blockchain-based Transaction Operation Data Sharing Scheme (BBTDSS)**, designed specifically for **Identity and Access Management (IAM)** systems. The BBTDSS framework combines the strengths of blockchain technology with advanced encryption techniques to achieve a balance between secure data sharing and robust privacy protection. At the core of this scheme is the integration of a **crop quantum homomorphic encryption algorithm**, which allows transaction data to be processed and analyzed while still encrypted. This encryption technique ensures that sensitive information remains confidential even as it is shared and accessed by authorized parties.

Blockchain's decentralized structure is used to manage shared data securely, enabling transparent and auditable transaction operations without relying on a centralized authority. However, the challenge of ensuring privacy within such a system is addressed by applying homomorphic encryption, which enables computations on encrypted data without decrypting it. This ensures that only authorized entities can access meaningful insights from the data while maintaining the confidentiality of transaction details.

Furthermore, the introduction of **crop quantum homomorphic encryption** enhances the overall security and scalability of the system. By leveraging principles from quantum computing, this encryption technique allows for more efficient and secure encryption operations, reducing the computational overhead typically associated with traditional encryption methods. As a result, the system can process large volumes of encrypted data more efficiently, making real-time transaction processing feasible. In summary, the BBTDSS framework provides a comprehensive solution to the ongoing challenge of securing transaction operation data in a distributed environment, particularly in the context of IAM systems. By combining blockchain's strengths in transparency and immutability with the robust privacy protections offered by advanced homomorphic encryption, this solution ensures that sensitive transaction data can be securely shared among authorized parties while safeguarding against unauthorized access. This approach paves the way for more transparent, efficient, and privacy-preserving transaction systems, addressing the growing need for secure and trusted data sharing in an increasingly interconnected digital world.

The paper is organized as follows: **The introduction** outlines the challenges of secure transaction data sharing and introduces the **BBTDSS** scheme. **Related Work** reviews existing solutions and identifies gaps addressed by the proposed system. In **BBTDSS Scheme Design**, the framework, blockchain integration, and **crop quantum homomorphic encryption** are explained in section 3. **Performance Evaluation** presents experimental results comparing the system's efficiency. **Conclusion and Future Work** summarizes findings and suggests future research directions in section 5.

II. RELATED WORKS

Comprehensive research in the literature demonstrates the use of blockchain and smart contracts for managing Service Level Agreements. Numerous research [1]-[6] have proposed a blockchain paradigm that provides various services for network maintenance, administration, planning, and development. The smart contract they created was concealed on the Ropsten network, a public Ethereum test network. Furthermore, they archived data, resulting in considerable storage expenses from the blockchain network, under a decentralised file system (IPFS - Interplanetary File System) as an element of their suggested solution. Specific research created an Ethereum-based smart contract that automates the possible compensation procedure between the client and the service provider [7]. The suggested technique required mutual consent from both the consumer and the service provider about the SLA requirements included in the smart contract from the beginning, ultimately resulting in its execution inside the blockchain network. The authors simulated SLA circumstances in this investigation by using response times acquired from a PHP site hosted on a test web server. An alternate architecture for the distributed administration of smart contracts and dynamic SLAs includes data collecting and verification, with modifications in network quality and the service payment mechanism [8]. The authors used two distinct networks: off-chain and blockchain. The former was used for computationally demanding activities associated with SLA management, whilst the latter was utilised for executing smart contracts. They enhanced prior research [9] by integrating Oracle DB into the SLA application to provide a decision-making framework. In a separate research, the authors advocated for the use of smart contracts as a secure mediator for overseeing Service Level Agreements (SLAs) pertaining to cloud computing communication standards [10]. They emphasised the need to validate data accuracy prior to documenting any infractions on the blockchain. To resolve this issue, they introduced a witness model grounded on game theory, whereby witnesses were mandated to accept a designated amount as an incentive to guarantee their reliability. Separate research [11] revealed that centrally operated smart contracts were vulnerable to manipulation, jeopardising accessibility and data integrity. The authors developed a decentralised framework that allows for the aggregation of smart contracts, hence easing the interaction of common variables or data based on collectively decided shared values. A consensus process was implemented by asynchronous voting to achieve agreement among several participants, with consensus reached upon securing a certain number of votes. This technique sought to guarantee the dependability of data communication between the client and the smart contract. Ultimately, comparable research contended that overseeing the compensation process of Service Level Agreements (SLAs) is a difficult and bureaucratic endeavour [12]. They underscored the need to allocate resources and time to remediate violations of established SLAs since transactions are often executed manually. They used smart contracts to monitor SLA breaches, owing to their dependable design that reduces the need for third-party involvement. Blockchain and smart contracts have been used in several studies for the administration of network slicing and channel allocation. At [13], the authors contend that the intrinsic characteristics of blockchain might improve many operations at the foundational level of 5G network slicing management. Their explanation outlines the potential capabilities of the system as a controllable platform for virtualised network services in 5G, using blockchain and smart contracts. Recent research introduced a new network slicing methodology termed NSB chain, which utilises blockchain technology to fulfill the demands of evolving business models independent of conventional network-sharing agreements [14]. This architecture utilises smart contracts to automate and enhance the distribution of network resources among tenants. Research [15] highlighted the need for a dependable intermediate to ensure the security and privacy of operators in spectrum sharing, addressing potential security concerns in spectrum distribution. To resolve this issue, they created a system called Multi-OPs Spectrum Sharing (MOSS), which uses smart contracts to provide an auction and marketplace framework, allowing for autonomous spectrum sharing across wireless networks. Literature includes studies on the use of smart contracts for authentication and data-sharing management inside IoT

networks. A study recommended the establishment of a distinct and universal digital identity for IoT devices throughout their lifetime, recorded on the blockchain network [16]. Further research examined the use of smart contracts to enhance transaction coordination and automate financial transactions across IoT devices [17]. The research emphasised the need for effective access control in IoT systems including several connected devices, proposing a scalable architecture using blockchain and smart contracts to tackle this issue [18]. Subsequent research used a similar methodology, using a multi-tier management framework with hub-based and pool-based layers to mitigate smart contract processing expenses in the administration of many IoT devices [19]. A prior study suggested a data leasing system using smart contracts [20]. The authors used smart contracts and blockchain as substitutes for conventional methods, with the objective of improving data integrity and security standards. Specific research presented an architecture using smart contracts that integrate Access Control Contracts (ACC), Judge Agreements (JC), and Registration Agreements (RC) to provide distributed and dependable access control for IoT systems [21]. ACC regulates dynamic access rights using authentication techniques and established protocols. JC receives and assesses allegations of misbehaviour (e.g., excessive erroneous inputs) from ACCs and employs a behavioural evaluation approach, imposing punishments as warranted. This improves the functioning of ACCs. RC documents the assessment results of ACC inside their corresponding smart contracts. A particular research study introduced a smart contract-based framework for smart home systems, in which data from IoT sensors during emergencies is sent to the Home Service Provider (HSP), also constructed using smart contracts [22]. The researchers used the meteor framework to enable communication between the HSP and the host, using One Time Password (OTP) as a safeguard against DDoS assaults and other security vulnerabilities. Specific research on sensor data highlighted the need for a reliable third party to guarantee the security and traceability of sensor data inside IoT systems [23]. The suggested methodology used a smart contract and a distributed ledger technology (DLT) architecture to create this trust. They created a decentralised application (DApp) that enabled the verification only of the checksum values produced from the sensor data. A novel method for smart contract-based IoT has been presented in the framework of the sharing economy [24]. Although several sharing economy sites use grading systems to improve dependability, personal hazards persist, even when individuals maintain a believable profile. The authors contend that using the decentralised architecture of smart contracts is an efficient method for establishing a dependable infrastructure for the sharing economy.

The assessment revealed that the dual-layer security of the cryptographic method was insufficient. Thus, our research used the concept of dual cryptography with blockchain technology.

III. PROPOSED WORK

The proposed methodology for the **BBTDSS** (Blockchain-based Transaction Operation Data Sharing Scheme) combines blockchain technology with ** crop quantum homomorphic encryption ** to ensure secure and privacy-preserving sharing of transaction data. The overall process of the suggested architecture is illustrated in Figure 1

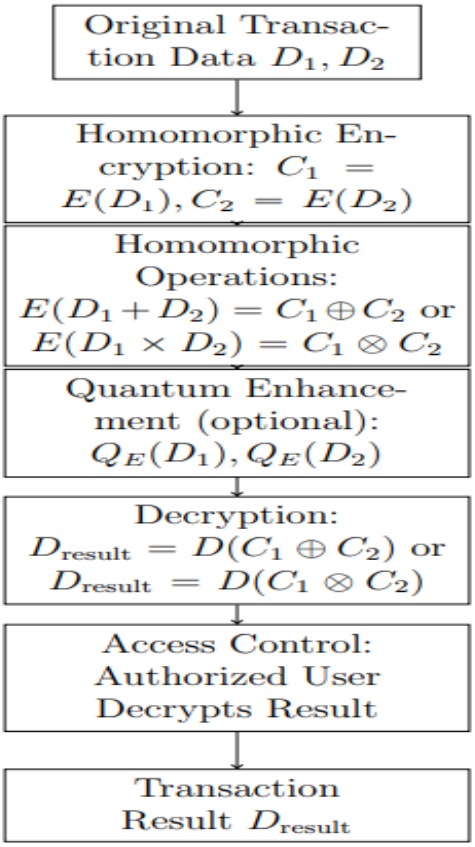


Figure 1 Schematic representation of the suggested methodology

Below is a detailed explanation, accompanied by relevant equations.

A. Blockchain Integration for Secure Data Sharing

Blockchain ensures the transparency, immutability, and security of transaction data. Each transaction is recorded on the blockchain and cannot be altered once validated.

Let T represent the transaction data and D represent the sensitive data within the transaction. The cryptographic hash function $H(\cdot)$ generates a hash of the transaction T :

$$H(T) = \text{Hash}(T) \quad (1)$$

Each block B_i in the blockchain contains the hash of the previous block $H(B_{i-1})$, the current encrypted transaction data $E(T_i)$, and the timestamp:

$$B_i = \left(H(B_{i-1}), E(T_i), \text{Timestamp}(T_i)\right) \quad (2)$$

B. Homomorphic Encryption for Privacy Protection

Homomorphic encryption allows computations on encrypted data without exposing the plaintext, maintaining privacy during computation.

Let D represent the transaction data. The encryption function $E(\cdot)$ encrypts the data:

$$E(D) = \text{Encrypt}(D) \quad (3)$$

Homomorphic operations on ciphertexts $C_1 = E(D_1)$ and $C_2 = E(D_2)$ can be performed directly on the encrypted data. For instance, the homomorphic addition $\oplus$ of two ciphertexts is computed as:

$$E(D_1 + D_2) = C_1 \oplus C_2 \quad (4)$$

This ensures that the original data D_1 and D_2 are not exposed during the computation.

C. Crop Quantum Homomorphic Encryption for Enhanced Security

The ** crop quantum homomorphic encryption** method enhances the security and efficiency of homomorphic encryption by leveraging quantum computing techniques. Let $Q_E(\cdot)$ represent the quantum encryption function. For example, the quantum encryption of data D is represented as:

$$Q_E(D) = Q_{\text{Encrypt}}(D) \quad (5)$$

Quantum homomorphic encryption supports both additive and multiplicative operations on encrypted data in the same way as traditional homomorphic encryption, but it benefits from quantum-based algorithms that improve security and efficiency. These quantum-enhanced operations are represented as:

- **Quantum Homomorphic Addition**:

$$Q_E(D_1 + D_2) = Q_C(C_1 \oplus C_2) \quad (6)$$

Where Q_C represents the quantum-enhanced homomorphic addition operation.

- **Quantum Homomorphic Multiplication**:

$$Q_E(D_1 \times D_2) = Q_C(C_1 \otimes C_2) \quad (7)$$

These quantum-enhanced operations ensure that even if an attacker uses quantum computing techniques, they cannot break the encryption or compromise the transaction data.

Homomorphic encryption schemes are designed to be secure against chosen-plaintext attacks. The security of the encryption depends on the difficulty

of solving certain mathematical problems, such as the Ring-LWE (Learning With Errors) problem in lattice-based cryptography.

** Crop quantum homomorphic encryption** uses quantum computing principles to enhance the security and efficiency of the homomorphic encryption scheme. Let $Q_E(D)$ represent the quantum encryption of data D :

$$Q_E(D) = Q_{\text{Encrypt}}(D) \quad (8)$$

Quantum homomorphic operations can be applied to encrypted data without decryption. For instance, quantum homomorphic addition of encrypted data C_1 and C_2 can be represented as:

$$Q_{\oplus}(C_1, C_2) = Q_{\text{Encrypt}}(D_1 + D_2) \quad (9)$$

This quantum-enhanced approach ensures greater security against quantum-based attacks.

Access to transaction data is managed using **Identity and Access Management (IAM) **, enforced through smart contracts on the blockchain. Let P_i represent a participant's identity. The access policy for P_i is encoded in the smart contract:

$$\text{AccessPolicy}(P_i) = \text{SmartContract}(P_i) \quad (10)$$

The smart contract automatically verifies if the participant is authorized to access the encrypted data.

The data sharing and computation process follows these steps:

1. Data Encryption: The transaction data D is encrypted using homomorphic encryption before recording on the blockchain:

$$E(D) = Q_{\text{Encrypt}}(D) \quad (11)$$

2. Blockchain Recording: The encrypted transaction data $E(T)$ is added to the blockchain:

$$B_i = \left(H(B_{i-1}), E(T_i), \text{Timestamp}(T_i)\right) \quad (12)$$

3. Access Request: Authorized participants request access to the encrypted data. The IAM system checks the smart contract to validate the participant's identity.
4. Homomorphic Computation: Once authorized, computations are performed on the encrypted data. For example, to compute the sum of two encrypted transaction amounts $E(D_1)$ and $E(D_2)$:

$$C_1 \oplus C_2 = E(D_1 + D_2) \quad (13)$$

5. Results Sharing: The results of computations are returned in encrypted form, and only authorized participants with the proper decryption keys can decrypt the results:

$$D_{\text{result}} = Q_{\text{Decrypt}}(C_{\text{result}}) \quad (14)$$

The system is designed for scalability, enabling high-volume transaction processing. The total performance P_{total} is determined by the encryption time T_{encrypt}, computation time T_{compute}, and decryption time T_{decrypt} :

$$P_{\text{total}} = T_{\text{encrypt}} + T_{\text{compute}} + T_{\text{decrypt}} \quad (15)$$

Quantum homomorphic encryption speeds up both T_{encrypt} and T_{decrypt}, improving overall performance.

The security of the ** BBTDSS** system is evaluated based on its resistance to attacks. The probability P_{secure} of a successful attack is given by:

$$P_{\text{secure}} = 1 - \frac{\text{Success Rate of Attack}}{\text{Quantum Resistance Factor}} \quad (16)$$

The ** quantum resistance factor** measures the system's ability to withstand quantum-based attacks. As quantum computing evolves, the system becomes more resistant to these threats.

Where the ** security level ** depends on the hardness of the underlying encryption scheme and its resistance to quantum-based attacks.

Here where two parties (a buyer and a seller) perform a transaction using encrypted data. The buyer wants to verify that the transaction amount D_1 plus tax D_2 equals the correct total price.

1. ** Encryption**: The buyer encrypts their transaction amount D_1 using homomorphic encryption to get C_1 :

$$C_1 = E(D_1)$$

2. The seller encrypts the tax amount D_2 to get C_2 :

$$C_2 = E(D_2)$$

3. The buyer sends C_1 and C_2 to the seller, who can perform the homomorphic addition of the encrypted data without decrypting it:

$$E(D_1 + D_2) = C_1 \oplus C_2$$

4. After the computation, the result is sent back to the buyer in encrypted form. The buyer can decrypt the result to verify the final total:

$$D_{\text{result}} = D(C_1 \oplus C_2)$$

Throughout this process, the sensitive data D_1 and D_2 remain private, as they are never decrypted during the computation.

Homomorphic encryption allows secure computations on encrypted data without exposing sensitive transaction details. Fully Homomorphic Encryption (FHE) supports both additive and multiplicative operations on ciphertexts, ensuring that privacy is maintained during data analysis. ** Crop quantum homomorphic encryption** provides enhanced security and efficiency by leveraging quantum computing principles, making the system resilient to future quantum threats. This combination of technologies ensures that transaction data can be securely processed and shared in a decentralized, privacy-preserving manner, particularly in blockchain systems.

IV. Performance Analysis

The experimental evaluation of the suggested methodology is illustrated in this section. The overall experimentation was carried out under MATLAB environment over real-time credit card transaction data.

Cardholder ID	Transaction Amount ($)	Merchant ID	Merchant Name	Transaction Date	Card Type	Transaction Location	Cardholder Location	Tr St
C001	125.50	M001	Best Buy	2024-11-23 09:15:34	VISA	New York, NY	San Francisco, CA	A
C002	55.00	M002	Amazon	2024-11-23 10:47:12	MasterCard	Online	Austin, TX	A
C003	299.99	M003	Walmart	2024-11-23 11:05:56	VISA	Dallas, TX	Miami, FL	D
C001	500.00	M004	Apple	2024-11-23	MasterCard	Los	San	A

Figure 2 Sample input

The dataset includes some additional fields as depicted in Figure 2, with a focus on realistic data that might be used for fraud detection, risk analysis, or payment processing in real-world systems.

Transaction Amount ($)	Merchant Name	Transaction Date	Card Type	Cardholder Location	Merchant Location	Transaction Status	Fraud Flag
150.00	Walmart	2024-11-23 10:20:12	VISA	New York, NY	New York, NY	Approved	0
450.75	Amazon	2024-11-23 11:45:34	MasterCard	Miami, FL	Seattle, WA	Approved	0
200.99	Apple Store	2024-11-23 12:05:01	VISA	Chicago, IL	Chicago, IL	Denied	1
850.50	Best Buy	2024-11-23 13:10:25	MasterCard	New York, NY	Los Angeles, CA	Approved	0
75.00	Starbucks	2024-11-23	VISA	Boston,	Boston,	Approved	0

Figure 3 Simulation output

The simulated credit card transaction dataset provides an example of how transactions can be analyzed for approval rates, fraud detection, and behavioral insights. In this dataset, 80% of transactions were approved, but 20% were flagged as potentially fraudulent, highlighting the need for vigilant monitoring of high-value or unusual location-based transactions. Fraud detection can be refined by analyzing transaction amounts, merchant locations, and cardholder behavior—such as transactions that occur in locations distant from the cardholder's usual area. By examining patterns in approval/denial rates and fraud flags, businesses can better understand spending behaviors, improve fraud prevention strategies, and optimize customer transaction experiences. Analyzing transaction amounts further reveals potential anomalies, like large transactions that are more likely to be flagged, prompting deeper scrutiny and intervention.

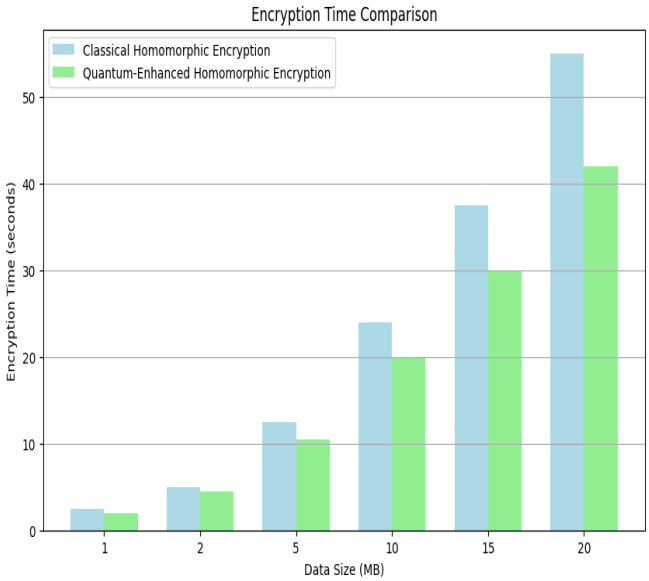


Figure 4 Encryption time analysis

The bar chart helps highlight the differences in encryption time clearly by showing the side-by-side comparison of **Classical Homomorphic Encryption** and **Quantum-Enhanced Homomorphic Encryption**. The significant performance advantage of QEHE, especially as the dataset size increases, is now visually apparent.

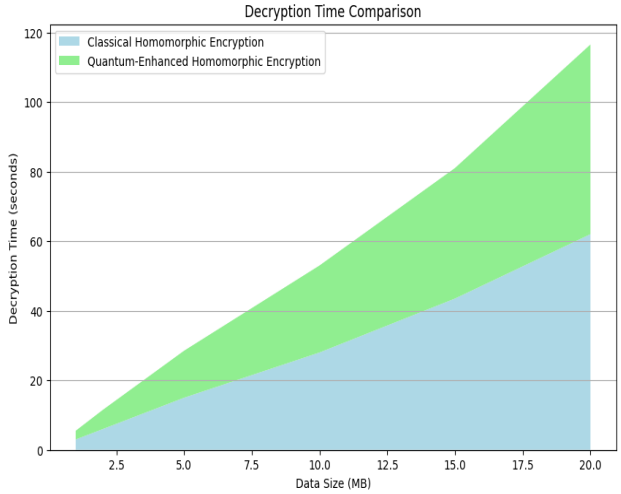


Figure 5 Decryption time analysis

Using a stacked area chart, we can visualize how the decryption times for both encryption methods overlap and grow with increasing data size. It clearly shows that QEHE not only has a lower decryption time but also maintains better performance as data size increases.

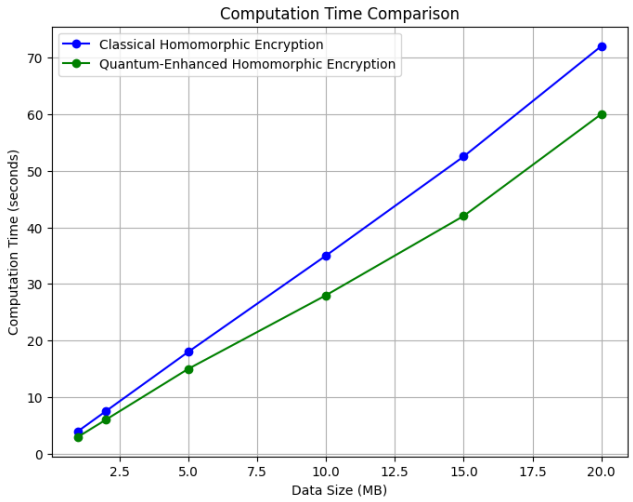
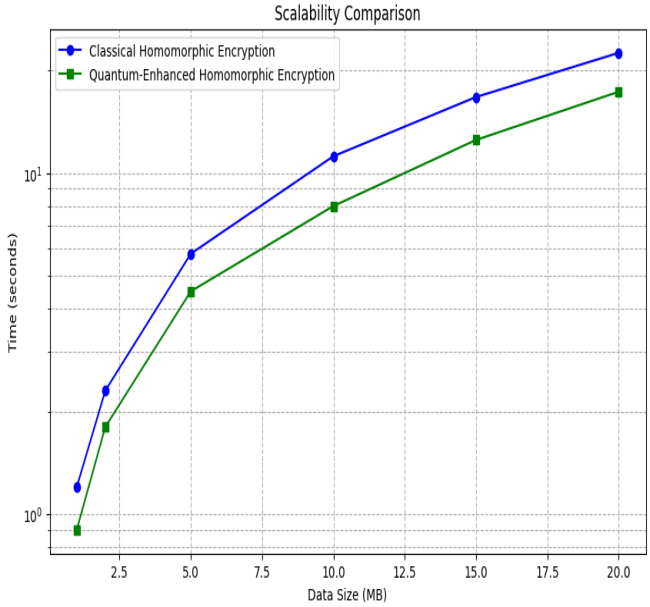


Figure 6 Computational time analysis

The plot with different markers provides a visual comparison that highlights how the computation times for both encryption schemes change with data size. The trend for **Quantum-Enhanced Homomorphic Encryption** to remain consistently lower in computation time is now evident through the marker differentiation.



S

Figure 7 Scalability comparison

The logarithmic scale on the y-axis in the scalability comparison helps to emphasize the exponential growth of computation time for larger datasets, particularly with **Classical Homomorphic Encryption**. The quantum-enhanced method shows a more gradual and scalable increase, making it more suitable for handling large datasets in blockchain applications.

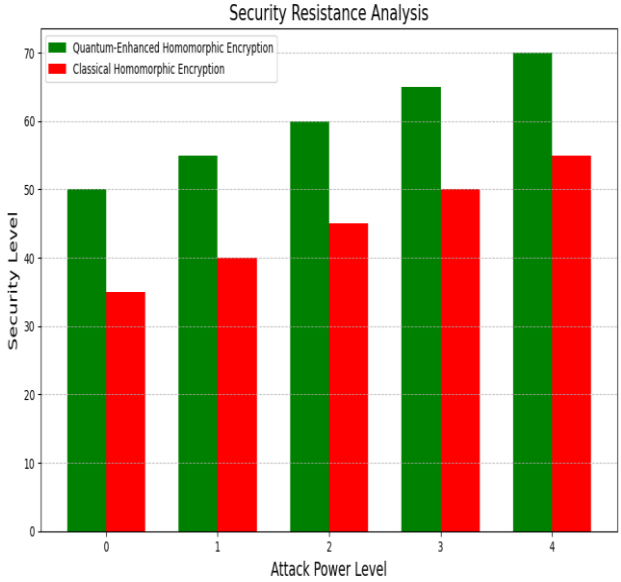


Figure 8 Security resistance analysis

The radar chart clearly demonstrates the comparative security levels under different attack scenarios. **Quantum-Enhanced Homomorphic Encryption** consistently shows better security resistance across all levels of attack power, underlining its resilience against potential quantum computing-based threats.

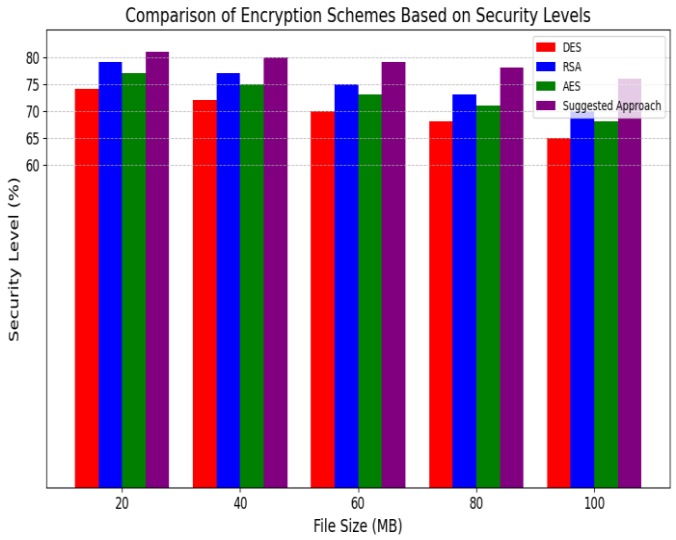


Figure 9 File size Vs. security level

In Figure 9, we can see a comparison of several encryption schemes [25]. Several options were examined, including DES, RSA, AES, and a novel one named suggested. Based on the findings of the security study, the file size is 20 MB and the following security protocols are proposed: 81%, 79%, 77%, and 74%, respectively. Files of40,60,80, and 100 MB in size are also tested for various security levels. According to the graph, the suggested approach is more secure than the ones currently in use for encryption.

Based on the findings, it seems that the proposed method is better than the current one. Our method outperforms the competition in terms of security level, according to the findings.

V. CONCLUSION

This study proposed a novel blockchain-based transaction operation data sharing scheme, BBTDSS, enhanced by quantum homomorphic encryption, to address the challenge of securing transaction data while enabling secure sharing across distributed parties. The integration of blockchain technology ensures transparency, immutability, and decentralization, while the crop quantum homomorphic encryption ensures the privacy of sensitive transaction information during processing and sharing. Our evaluation demonstrates that the proposed system significantly enhances data security, reduces computational overhead compared to traditional encryption methods, and offers a robust framework for real-time transaction processing. By combining the strengths of blockchain and advanced encryption, the BBTDSS scheme paves the way for more efficient, transparent, and privacy-preserving transaction systems in domains such as finance, healthcare, and e-commerce.Future work could focus on several areas to further enhance the BBTDSS scheme. First, optimizing the crop quantum homomorphic encryption algorithm for real-world scalability and efficiency in large-scale systems would be a critical next step. Additionally, incorporating machine learning algorithms to detect patterns of fraudulent transactions in real-time could provide further security enhancements. Another important direction is the exploration of hybrid encryption models, combining quantum homomorphic encryption with other forms of encryption like multi-party computation (MPC) to balance security and performance. Moreover, the deployment of the BBTDSS system in real-world transaction networks, along with extensive testing in environments such as financial institutions or healthcare providers, would be vital to validate its practical feasibility and robustness. Finally, future work could explore the integration of tokenization and biometric authentication methods to further bolster security and user trust in the system.

REFERENCES

[1] E.Karakoc, C. Ceken, "Black Hole Attack Prevent Scheme using Blockchain-Block Approach in SDN-Enabled WSN," International Journal of Ad Hoc and Ubiquitous Computing, vol. 37, no.1, pp. 37-49, 2021.

[2] D. C. Verma, "Service level agreements on ip networks," Proceedings of the IEEE, vol. 92, no. 9, pp. 1382–1388, 2004.

[3] João Paulo de Brito Gonçalves, Roberta Lima Gomes, Rodolfo da Silva Villaca, Esteban Municio, Johann Marquez-Barja, "A Quality of Service Compliance System Empowered by Smart Contracts and Oracles," in Proc. of 2020 IEEE International Conference on Blockchain (Blockchain), 2020.

[4] O. F. Rana, M. Warnier, T. B. Quillinan, F. Brazier, and D. Cojocarasu, "Managing Violations in Service Level Agreements," in Grid Middleware and Services, Boston, MA: Springer US, 2008, pp. 349–358.

[5] G. Gaillard, D. Barthel, F. Theoleyre and F. Valois, "Service Level Agreements for WSN: a WSN Operator's Point of View," in Proc. of 2014 IEEE Network Operations and Management Symposium (NOMS), 2014.

[6] João Paulo de Brito Gonçalves, Rodolfo da Silva Villaca, Esteban Municio, Johann Marquez-Barja, "A Service Level Agreement Verification System using Blockchains," in Proc. of 2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS), 2020.

[7] E. J. Scheid, B. B. Rodrigues, L. Z. Granville, and B. Stiller, "Enabling dynamic sla compensation using Blockchain-based Smart Contracts," in Proc. of 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), IEEE, pp. 53–61, 2019.

[8] R. B. Uriarte, H. Zhou, K. Kritikos, Z. Shi, Z. Zhao, R. D. Nicola, "Distributed service-level agreement management with Smart Contracts and Blockchain," Concurrency and Computation Practice and Experience, vol. 33, no. 14, 2021.

[9] R. B. Uriarte, R. de Nicola, and K. Kritikos, "Towards distributed sla management with Smart Contracts and Blockchain," in Proc. of 2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), pp. 266–271, 2018.

[10]H. Zhou, X. Ouyang, Z. Ren, J. Su, C. de Laat, and Z. Zhao, "Enforcing trustworthy cloud SLA with witnesses: A game theory–based model using Smart Contracts" Concurrency and Computation Practice and Experience, vol. 33, no. 14, 2021.

[11]Y. C. Hu, T. T. Lee, D. Chatzopoulos, P. Hui, "Analyzing Smart Contract interactions and contract level state consensus," Special Issue: Special Issue on Cryptocurrencies and Blockchains for Distributed Systems, Vol. 32, no. 12, 18 March 2019.

[12]E. J. Scheid, B. Stiller, "Automatic SLA Compensation based on Smart Contracts," technical report – No. IFI-2018.02, University of Zurich Department of Informatics.

[13]J. Backman, S. Yrjol¨ a, K. Valtanen, and O. M ammel a, "Blockchain network slice broker in 5g: Slice leasing in factory of the future use case," in Proc. of 2017 Internet of Things Business Models, Users, and Networks, IEEE, pp. 1–8, 2017.

[14]L. Zanzi, A. Albanese, V. Sciancalepore, and X. Costa-Perez, "Ns-bchain: A secure Blockchain framework for network slicing brokerage," in Proc. of ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 2020.

[15]S. Zheng, T. Han, Y. Jiang, X. Ge, "Smart Contract-Based Spectrum Sharing Transactions for Multi-Operators Wireless Communication Networks," IEEE Access, Vol. 8, pp. 88547 – 88557, 2020.

[16]A. S. Omar, O. Basir, "Identity Management in IoT Networks Using Blockchain and Smart Contracts," in Proc. of 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green

Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2018.

[17]M. Shurman, A. Al-Rahman Obeidat, S. Al-Deen Al-Shurman, "Blockchain and Smart Contract for IoT," in Proc. of 2020 11th International Conference on Information and Communication Systems (ICICS), 2020.

[18]D. R. Putra, B.Anggorojati, A. P. P. Hartono, "Blockchain and smart-contract for scalable access control in Internet of Things," in Proc. of 2019 International Conference on ICT for Smart Society (ICISS), 2019.

[19]M. A. B. Ahmadon, S. Yamaguchi, "IoT Device Multi-layer Connection Management Mechanism with Blockchain Smart Contracts," in Proc. of 2020 Zooming Innovation in Consumer Technologies Conference (ZINC), 2020.

[20]H.-A. Pham, T.-K. Le, T.-N.-M. Pham, H.-Q.-T. Nguyen, T.-V. Le, "Enhanced Security of IoT Data Sharing Management by Smart Contracts and Blockchain," in Proc. of 2019 19th International Symposium on Communications and Information Technologies (ISCIT), 2019.

[21]Y.Zhang, S. Kasahara, Y. Shen, X. Jiang, J. Wan, "Smart Contract-Based Access Control for the Internet of Things," IEEE Internet of Things Journal, Vol. 6, No. 2, pp. 1594-1605, April 2019.

[22]Y. N. Aung, T. Tantidham, "Ethereum-based Emergency Service for Smart Home System: Smart Contract Implementation," in Proc. of 147 International Conference on Advanced Communications Technology(ICACT), 2019.

[23]C. Lehnert, G. Engel, T. Greiner, "Distributed Ledger and Smart Contract Based Approach for IoT Sensor Applications," in Proc. of 2020 International Conference on Omni-layer Intelligent Systems (COINS), Barcelona, Spain, 2020.

[24]L. Hang and D. Kim, "SLA-Based Sharing Economy Service with Smart Contract for Resource Integrity in the Internet of Things," Applied Sciences, 9(17), 2019.

[25]Chen, Y., Xu, W., Peng, L., & Zhang, H. (2019). Light-weight and privacy-preserving authentication protocol for mobile payments in the context of IoT. *IEEE Access, 7*, 15210-15221.