Towards a Decentralized and Privacy-Preserving Quality of Experience System for 6G Networks

RODRIGO DUTRA GARCIA*, Institute of Mathematics and Computer Science, University of São Paulo, Brazil

GOWRI RAMACHANDRAN, Faculty of Science, Queensland University of Technology, Australia

CHRISTIAN ESTEVE ROTHENBERG, School of Electrical and Computer Engineering, Universidade Estadual de Campinas, Brazil

DANIEL MACÊDO BATISTA, Department of Computer Science, University of São Paulo, Brazil

JÓ UEYAMA, Institute of Mathematics and Computer Science, University of São Paulo, Brazil

The transition to 6G networks is expected to support a broader range of user-centric applications. As these applications expand, Quality of Experience (QoE) has emerged as a key metric for evaluating user satisfaction. However, the use of centralized systems lacks transparency and limits users' ability to govern their data usage. It also introduces challenges in managing QoE data while preserving privacy. At the same time, verification mechanisms are needed that allow regulators to evaluate compliance without exposing confidential business information. To address this, we propose a decentralized, privacy-preserving QoE system that integrates blockchain with fully homomorphic encryption (FHE). This design enables transparent evaluations between users and service providers by supporting computations directly on encrypted data, ensuring that all information remains protected throughout the process. Users contribute QoE metrics through a decentralized infrastructure and retain control over their data. Regulators can monitor compliance without accessing raw data, and service providers can use encrypted QoE data to perform privacy-preserving computations via smart contracts without relying on a central authority. We developed a proof-of-concept integrating FHE smart contracts compatible with Ethereum Virtual Machine (EVM) blockchains and evaluated their performance using a video streaming dataset. Our results show that encryption and FHE operations consistently occur within milliseconds when tested in a local environment. We also evaluated these operations on a public blockchain testnet. In this setting, our system adds a millisecond-scale delay while supporting privacy-preserving computations directly on encrypted user data.

Additional Key Words and Phrases: Blockchain, Quality of Experience, Privacy, Encryption, Decentralization

1 INTRODUCTION

The development of Beyond 5G (B5G) and 6G networks is expected to deliver faster connectivity, supporting new applications and multimedia services [46, 82]. These networks will likely be heterogeneous, densely deployed, and highly dynamic [28, 83], enabling advanced digital services such as virtual reality (VR), augmented reality (AR),

Authors' Contact Information: Rodrigo Dutra Garcia, rgarcia@usp.br, Institute of Mathematics and Computer Science, University of São Paulo, São Carlos, São Paulo, Brazil; Gowri Ramachandran, Faculty of Science, Queensland University of Technology, Brisbane, Queensland, Australia, g.ramachandran@qut.edu.au; Christian Esteve Rothenberg, chesteve@dca.fee.unicamp.br, School of Electrical and Computer Engineering, Universidade Estadual de Campinas, Campinas, São Paulo, Brazil; Daniel Macêdo Batista, batista@ime.usp.br, Department of Computer Science, University of São Paulo, São Paulo, São Paulo, Brazil; Jó Ueyama, Institute of Mathematics and Computer Science, University of São Paulo, São Carlos, São Paulo, Brazil, joueyama@icmc.usp.br.

ACM 2769-6480/2025/10-ART
https://doi.org/10.1145/3772072

and immersive holography. These services require not only high bandwidth but also ultra-low latency [24, 88]. As user-centric applications expand, quality of experience has become an important metric [88] for service providers. Unlike traditional quality of service (QoS), which emphasizes technical parameters like latency and throughput, QoE captures overall user satisfaction by combining objective performance metrics with subjective perceptions influenced by cognitive, perceptual, and behavioral factors [13, 55]. However, the increasing focus on QoE in 6G raises privacy concerns, particularly as sensor-driven connectivity expands and human-centric metrics are adopted as key performance indicators in 6G communications [24, 64].

Since QoE is a user-centered measure, it is commonly evaluated using direct feedback methods, such as surveys and Mean Opinion Score (MOS) ratings [17]. Another approach involves analyzing sensitive data, including psychological indicators like facial expressions recorded by sensors [1, 11]. These methods have been used to evaluate QoE in virtual reality under varying network conditions [78, 88] and in multimedia streaming [7, 17]. Different technologies like Content Distribution Networks [89], Edge Computing [87], and Edge Caching [60, 94] and Machine Learning approaches [51] are used to improve user experience. These systems often rely on cloud computing and centralized servers for data storage and processing. However, this dependency raises privacy concerns, as users are unable to verify how their data is accessed, used, or shared by service providers and third parties. The lack of transparency limits users' ability to govern data usage and verify that providers deliver contracted service levels. As 6G networks are expected to support demanding applications with strict QoE requirements, both transparency and privacy are required. Service providers require greater visibility into the overall quality of experience and network conditions, including detailed and traceable network data, to improve service delivery to the users. At the same time, verification mechanisms are needed that allow regulators to verify compliance without disclosing proprietary or confidential business information.

While QoE personalization enhances user experience, it also creates trade-offs with data privacy [31], requiring mechanisms that address both sides: allowing service providers to evaluate needs and satisfaction levels across the users, while simultaneously giving users granular control over access to their data, all within end-to-end privacy. There is a need for a privacy-preserving QoE mechanism that maintains user data in encrypted form throughout all operations performed by service providers and network operators, particularly for 6G networks where emerging applications demand data protection. Current privacy-preserving methods demonstrate specific limitations, particularly regarding data processing while keeping data confidential. Federated Learning (FL), while protecting raw data, introduces communication overhead and model performance trade-offs [34, 44]. Research identifies FL vulnerabilities to model poisoning [93], inference attacks [75], and model inversion attacks [90]. Machine learning models at network edges introduce security vulnerabilities through potential attack surfaces and data exposure risks [32, 65]. The analysis of encrypted traffic for QoE evaluation faces a core constraint: while traditional methods often require decryption, potentially compromising privacy. Studies by Dillbary et al. [26] and Gutterman et al. [39] demonstrate that QoE metrics can be inferred from metadata or observable patterns in encrypted traffic, thereby preserving encryption and user privacy. However, it does not permit direct operations on encrypted data, which limits the ability to understand user requirements.

Differential Privacy methods inherently involve trade-offs between privacy protection and data utility. Esper et al. [30] demonstrated that increasing privacy protection, such as through Geo-Indistinguishability to protect the user's exact location, can lead to reduced offloading efficiency and increased latency in edge computing scenarios. Similarly, Feng et al. [31] showed that enhancing privacy levels by obfuscating user locations using DP mechanisms can degrade QoE and QoS in location-based services. In video streaming applications, Jin et al. [48] proposed a privacy protection module specifically designed for immersive video streaming environments, while Zhang et al. [101] demonstrated how privacy mechanisms create processing overhead that reduces performance. Jin et al. [48] and Esper et al. [30] established that stronger privacy measures decrease measurement accuracy, making it difficult to achieve both privacy protection and QoE evaluation simultaneously.

Unlike these approaches, this work employs fully homomorphic encryption, enabling computations directly on encrypted data without the need for decryption [36], preserving both privacy and the fidelity of fine-grained QoE metrics that can be lost through aggregation or noise injection. Complementing this, blockchain offers a decentralized and transparent infrastructure for storing and managing data access. Smart contracts enable automated validation and computation within these networks, eliminating the need for intermediaries or central servers [42]. The proposed system aims to address the following research questions:

(1) How does transparency in a decentralized infrastructure impact user trust and enable service providers to understand users' needs?
(2) How can end users govern data sharing and computations on their encrypted QoE data with authorized parties (such as service providers and regulators), enabling these parties to perform operations on the data while protecting both user privacy and business confidentiality?

1.1 Research Contributions

This research introduces a decentralized system for privacy-preserving computation that allows service providers to evaluate and enhance services by performing computations directly on encrypted QoE data. By combining the transparency and integrity of blockchain records with the deterministic properties of fully homomorphic encryption, the system ensures that user-authorized parties can verify the encrypted data's origin and the correctness of the computations without exposing user information. Regulators can assess compliance without infringing on user privacy or revealing companies' proprietary business data. The key contributions are as follows:

- A blockchain-based decentralized system that enables users to govern the sharing of multiple encrypted QoE metrics. Employing fully homomorphic encryption allows third parties to perform computations on the encrypted data without revealing the underlying information.
- A decentralized verifiable computation mechanism leveraging the transparency of the underlying blockchain and determinism of FHE operations. It allows different stakeholders to verify computations while the data remains encrypted.
- An evaluation of the proposed system using a real-world QoE dataset, demonstrating its feasibility and performance in processing multiple encrypted QoE metrics. The implementation uses the fhEVM [98] library, which is based on a Rust implementation of Fast Fully Homomorphic Encryption over the Torus (TFHE)[100], originally proposed by Chillotti et al. [20].

1.2 Research Outline

The organization of this study is as follows: Section 2 provides background on the quality of experience and the challenges in ensuring privacy and trustworthiness in these systems. Additionally, it introduces blockchain technology and homomorphic encryption. Section 3 reviews related works on privacy-preserving approaches, including homomorphic encryption, differential privacy, federated learning, and blockchain. Section 4 then introduces our proposed decentralized QoE system, which employs blockchain and fully homomorphic encryption. It also includes a threat analysis and security model. Section 5 presents the proof of concept evaluation using a real video streaming dataset. Section 6 examines how the proposed system addresses the research questions, its potential impact on QoE systems for 6G networks, and scalability challenges. Section 7 summarizes the study and outlines future work.

2 BACKGROUND

2.1 Blockchain on 6G Network Management

Blockchain (BC) technology incorporates cryptography, a peer-to-peer (P2P) network protocol, and a consensus mechanism. Unlike centralized systems, blockchain systems do not rely on a central server for data management.

Instead, all nodes in the network maintain the system. Users can submit transactions that are validated and maintained through consensus criteria among the nodes, providing transparency and data immutability by employing cryptography [97]. This decentralized approach makes it fault-tolerant and resistant to malicious attacks [38] and censorship [41]. Additionally, blockchain enables the development of smart contracts, which are self-executing programs stored on the blockchain. These contracts automatically execute predefined actions when specific conditions are met without the need for intermediaries.

Blockchain helps to address numerous complex security and management challenges in 6G, particularly around governance, where different stakeholders might have competing interests [74, 79, 102]. For instance, the BC decentralization and transparent infrastructure improve spectrum management efficiency and allow for better allocation of frequency resources across operators [6, 22]. It also enables seamless resource sharing and network slicing, providing secure management of isolated virtual networks. Blockchain can also be utilized in edge strategies to reduce latency in 6G networks [76, 91]. The immutable nature of records enhances security and regulatory compliance, establishing trust in multi-operator environments crucial for shared infrastructure scenarios. Furthermore, blockchain can support automated Service Level Agreements between various stakeholders in the 6G ecosystem through smart contracts [69]. Literature suggests that decentralized management is promising for handling a large number of IoT devices in 6G networks, as centralized IoT architectures face significant challenges, including high maintenance costs, vulnerability to malicious attacks, and scalability issues [66].

Despite the benefits of blockchain for 6G, there are challenges, such as privacy concerns due to large-scale heterogeneous networks and time-sensitive applications [33, 62]. Privacy leakage is a key concern in blockchain transactions and stored data, as heterogeneity in 6G networks raises questions about preventing the unauthorized disclosure of confidential and private information generated by user devices. Current methods rely on a third party (e.g., a cloud provider) to provide security. This creates a vulnerability for privacy if the cloud systems are attacked, potentially breaching participants' privacy [9, 52]. As 6G will prioritize user-centric applications, privacy protection mechanisms will be necessary for QoE management [46, 64, 81].

2.2 Quality of Experience

Quality of experience is an essential metric reflecting user satisfaction [4]. The International Telecommunications Union (ITU) defines QoE as "the overall acceptability of an application or service, as perceived subjectively by the end-user" [45]. QoE differs from QoS by emphasizing user satisfaction, integrating both objective performance metrics and subjective user-centered perceptions influenced by human factors, such as perceptual, cognitive, and behavioral interactions with the service [13, 55]. These methods account for personal and psychological dimensions of user experience, incorporating elements like facial expressions and heart rate to assess perceptual quality [1, 11].

Subjective approaches depend on direct input from users, including surveys, questionnaires, or rating mechanisms such as the MOS, providing a numerical evaluation of the perceived quality or satisfaction with a particular service or application [17]. QoE evaluation examples include the use of MOS ratings in VR environments under varying network conditions [78, 88], as well as assessments of multimedia streaming and video services [7, 17]. As digital services such as VR and immersive holographic applications become more common in the context of 6G, user experience emerges as a primary concern [24, 88], requiring future 6G networks to emphasize QoE rather than conventional QoS parameters. Researchers are actively working on methods to enhance QoE by centering on the end-user's perception [17, 54, 88].

6G networks are expected to be heterogeneous, user-centric, and designed to support diverse services and user needs [49]. They will deliver extreme performance through immersive communication, supporting a wide range of applications. Managing both QoS and QoE in this environment will be challenging due to the diversity of services, varying user expectations, and constantly evolving requirements [7]. QoE data management in 6G will increasingly rely on collecting and analyzing sensitive user data, including location, biometric information, and behavioral patterns, to personalize services and optimize performance [64]. As user experience becomes a key metric, maintaining privacy will remain a central concern. Users will demand greater transparency and control over how their data is collected, processed, and used. Privacy-preserving QoE mechanisms will be necessary to enable personalized service improvements while protecting sensitive information [64].

2.3 Fully Homomorphic Encryption

Fully homomorphic encryption is a cryptographic technique that allows unlimited mathematical operations to be performed directly on encrypted data without the need for decryption, ensuring that data remains protected throughout the entire processing [67]. FHE was first introduced by Rivest et al. [77] in 1978. However, a practical implementation of FHE was not achieved until 2009, when Gentry [36] proposed the first FHE scheme enabling the development of solutions focusing on privacy concerns [25, 37]. Unlike Partially Homomorphic Encryption (PHE), which permits only a single type of operation (either addition or multiplication), and Somewhat Homomorphic Encryption (SHE), which supports a finite number of operations, FHE enables a sequence of both additions and multiplications on encrypted data. This capability enables the development of applications with complex computations on sensitive data while maintaining privacy.

Figure 1 shows a generic process of FHE. The process begins with data owners, typically users, who encrypt their sensitive information using a public key (pk) before sharing it. This step ensures that the raw data remains confidential. Once the data is encrypted, other users or service providers can perform a wide range of computations directly on it. These homomorphic operations can include additions, multiplications, and even more complex functions. As homomorphic operations are performed, they introduce noise that accumulates over time, potentially affecting the accuracy of results. To address this, a bootstrapping step may be necessary. Introduced by Gentry [36], bootstrapping refreshes the ciphertext, reducing noise and allowing for further computations without compromising the encryption. The final step in the FHE process is decryption. Only at this point can the computations' results be revealed using a secret key (sk). Importantly, this decryption only exposes the final output, maintaining the privacy of all intermediate data and the original inputs throughout the entire process.

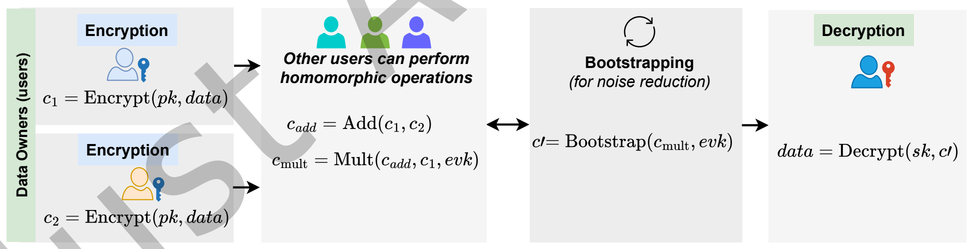
Data Owners (users)
Encryption
$c_1 = \mathrm{Encrypt}(pk, data)$
Encryption
$c_2 = \mathrm{Encrypt}(pk, data)$
Other users can perform homomorphic operations
$c_{add} = \mathrm{Add}(c_1, c_2)$
$c_{\mathrm{mult}} = \mathrm{Mult}(c_{add}, c_1, evk)$
Bootstrapping
(for noise reduction)
$c\prime = \mathrm{Bootstrap}(c_{\mathrm{mult}}, evk)$
Decryption
$data = \mathrm{Decrypt}(sk, c\prime)$

Fig. 1. Overview of the fully homomorphic encryption process. The diagram illustrates the key steps: (1) Data owners encrypt their sensitive information; (2) Other users or service providers perform homomorphic operations on the encrypted data; (3) Bootstrapping is applied for noise reduction when necessary; and (4) The final result is decrypted, revealing only the computed output while preserving the privacy of the original inputs and intermediate calculations.

In this research, we employed the fhEVM library of the Zama protocol, which uses a Rust implementation of Fast, Fully Homomorphic Encryption over the Torus (TFHE-rs) [100]. This library enables privacy-preserving computations on the EVM blockchains [98]. The encryption process uses a global FHE key as the public key to encrypt all inputs and private states [98]. This approach enables the combination of encrypted data from multiple users and across multiple smart contracts. For decryption, the protocol combines multi-party computation and the threshold scheme proposed by Dahl et al. [23]. The private decryption key is secret-shared among validators (i.e., network nodes), with no single party holding the complete key. When decryption is needed, a threshold protocol is executed where multiple validators collaborate to decrypt the data collectively. Smart contracts can decrypt values by calling a decrypt function, triggering the execution of a threshold protocol with a certain latency, resulting in a plain-text value of the corresponding data type [98]. This system divides the decryption key among multiple validators, ensuring that no single entity can access plain-text data independently. The threshold protocol requires collaboration from a minimum number of validators for decryption while maintaining operational integrity even when some validators are compromised or unavailable. The next section reviews the related works in privacy-preserving techniques, focusing on homomorphic encryption, differential privacy, and federated learning.

3 RELATED WORKS

The expansion of devices in future wireless networks will generate massive amounts of data, requiring decentralized approaches for processing and storage rather than relying on centralized servers [18]. Such decentralized systems can enhance transparency, enabling user trust and greater participation in network governance [79]. As QoE in 6G relies on extensive user data collection, ensuring privacy preservation is critical to maintain trust and compliance [64, 88].

3.1 Homomorphic Encryption Approaches

Homomorphic encryption (HE) has been adopted to enable secure computations on encrypted data across various domains, including IoT, healthcare, and machine learning. For instance, in the Internet of Vehicles (IoV), Xue et al. [95] developed a dynamic pricing scheme for mobile edge computing (MEC) that secures the offloading process using homomorphic encryption. This scheme ensures that user data remains private while allowing MEC servers to perform computations on encrypted data, protecting it from malicious servers and eavesdroppers.

In healthcare, Kumbhar and Srinivasa Rao [53] proposed a multi-key homomorphic encryption approach for secure healthcare data sharing in federated learning. The proposed approach generates optimal encryption keys, and a deep residual network classifies privacy-sensitive data. Local model updates are encrypted and aggregated into a global model. In the Internet of Things context, Hijazi et al. [43] presented a federated learning approach for IoT-enabled smart cities that integrates FHE to ensure privacy and security during model training. Also, the authors evaluated four FHE-based FL approaches where encrypted data is transmitted over secure channels.

In the context of machine learning, Wang et al. [92] proposed a framework that integrates HE with deep learning models to enable secure data processing in IoT applications. Their system achieved high accuracy on encrypted datasets, demonstrating the potential of HE in preserving data privacy without sacrificing model performance. To address scalability and efficiency challenges, Jastaniah et al. [47] introduced a multi-key HE solution for wearable data aggregation, offering both flexibility and privacy protection for resource-constrained devices like wearables. Sendhil and Amuthan [84] presented a quaternion-based HE scheme aimed at mitigating false data injection attacks in fog computing environments, enhancing privacy and fault tolerance in decentralized systems. Despite the HE features and applications, none of the works above explored the application of FHE in the context of quality of experience with decentralized infrastructure.

3.2 Differential Privacy Approaches

Differential privacy mechanisms have been employed to address privacy concerns in multimedia and network systems while maintaining service quality. In immersive video streaming, Jin et al. [48] introduced a framework integrating privacy protection, viewport prediction, and adaptive bitrate allocation. They applied differential privacy to add noise to viewport and gaze data, considering spatial and temporal correlations, and employed a transformer-based cross-modal attention model to predict future viewports using historical data and video content.

In vehicular environments, Feng et al. [31] developed a system to preserve location privacy for services such as navigation and parking. Their framework enables users to quantify the trade-off between privacy and utility through a QoE metric evaluated under varying privacy levels. For mobile edge networks, Zhang et al. [101] proposed the privacy-preserving Q-learning-based video caching framework (VC-PPQ). This approach combined local differential privacy to protect user locations and preferences with a data aggregation model that balances accuracy and privacy, thereby enhancing video Quality of Experience (VQoE). However, using differential privacy, stronger privacy guarantees result in reduced data utility due to noise addition and lower service performance. Additionally, there is a trade-off between the level of privacy and measurement accuracy.

3.3 Privacy-Preserving Federated Learning Approaches

As B5G/6G networks evolve, addressing user data privacy concerns has become increasingly necessary [51]. Researchers have proposed different decentralized, privacy-preserving approaches to tackle this challenge, with federated learning emerging as a prominent solution across different network applications. In video services, several studies have leveraged FL to enhance QoE while maintaining user privacy. Setayesh and Wong [85] proposed a THz-enabled 360° video streaming system with multi-antenna access points, combining personalized federated learning for viewport prediction and deep reinforcement learning for joint bitrate selection and beamforming. Likewise, Pu et al. [73] proposed a UHD video streaming framework for cloud-native 5G networks, integrating erasure-coded storage and multi-source streaming to improve QoE. Federated learning updates service quality metrics, combining user-side reinforcement learning for server selection with global aggregation.

Content caching has been another area where FL has shown significant promise in preserving privacy while improving network performance. Md. Fadlullah and Kato [68] proposed a privacy-preserving algorithm for 6G networks using FL, specifically targeting content caching optimization. Extending the application of FL in caching, Maale et al. [63] introduced a proactive caching scheme for UAV-assisted networks. Similarly, Li et al. [58] proposed a collaborative UAV caching and trajectory optimization framework to improve users' QoE in dynamic traffic environments. Using multi-agent deep reinforcement learning, UAVs learn caching and movement strategies from local observations.

Some studies have explored more advanced FL applications in the context of vehicular networks. Corcuera Bárcena et al. [21] employed FL with explainable AI models for QoE forecasting, addressing the need for interpretability in AI-driven QoE management. Taking a different approach, Li et al. [59] proposed a cooperative edge intelligence architecture using federated multi-agent reinforcement learning, demonstrating how FL can be combined with other AI techniques to create more sophisticated, privacy-preserving network management systems. Porcu et al. [72] introduced a cluster-based FL approach for QoE prediction, showing how FL can be optimized for improved performance in QoE modeling.

Researchers have also recognized the potential of blockchain in addressing limitations of centralized and federated learning approaches, particularly in enhancing the security and privacy of machine learning models [50, 61]. Nguyen et al. [70], introduced a decentralized architecture based on a Decentralized Autonomous Organization (DAO) to improve Federated Learning security in QoE estimation.

Table 1 summarizes the related works and the proposed system. While these studies demonstrate significant advancements in privacy-preserving QoE management using FL and differential privacy, they also reveal certain limitations. Users cannot transparently submit or view their satisfaction ratings as part of their QoE metrics while ensuring that their sensitive data remains private. Additionally, service providers cannot perform computations on user data to enhance their services without compromising user privacy, as doing so would require decrypting the data. To address these issues, we propose a novel approach that combines smart contracts with FHE. This decentralized infrastructure allows users to participate in the QoE evaluation process, sharing network metrics and service experiences while maintaining end-to-end privacy. Unlike existing approaches, our FHE-based system allows users to control their data usage while allowing service providers to perform computations directly on encrypted data. This enables QoE improvements without compromising user privacy. The next section introduces our proposed decentralized QoE system.

Table 1. Comparison of related works and the proposed system: The proposed system combines FHE with blockchain-based techniques to ensure end-to-end privacy of QoE data, support decentralized computation, and strengthen governance in multimedia services.

Work	Application	Focus on QoE	Blockchain-based	Privacy-Preserving Mechanism	Computation Directly on Encrypted User Data	Decentralized Computation Using Smart Contracts
[53]	Healthcare	No	No	Homomorphic Encryption	Yes	No
[95]	Mobile Edge Computing	No	No	Homomorphic Encryption	Yes	No
[43]	Internet of Things	No	No	Fully Homomorphic Encryption	Yes	No
[92]	Deep Learning	No	No	Homomorphic Encryption	Yes	No
[47]	Internet of Things	No	No	Partial Homomorphic Encryption	Yes	No
[84]	Fog Computing	No	No	Fully Homomorphic Encryption	Yes	No
[48]	Video Streaming	Yes	No	Differential Privacy	No	No
[31]	Location Based Services	Yes	No	Differential Privacy	No	No
[101]	Video Caching	Yes	No	Differential Privacy	No	No
[70]	QoE Estimation	Yes	Yes	Federated Learning	No	No
[85]	Video Streaming	Yes	No	Federated Learning	No	No
[73]	Video Streaming	Yes	No	Federated Learning	No	No
[58]	Content Caching	Yes	No	Federated Learning	No	No
[68]	Content Caching	Yes	No	Federated Learning	No	No
[63]	Content Caching	Yes	No	Federated Learning	No	No
[21]	QoE Forecasting	Yes	No	Federated Learning	No	No
[59]	Cooperative Edge Intelligence	Yes	No	Federated Learning	No	No
[72]	QoE Modelling	Yes	No	Federated Learning	No	No
Our Work	**Multimedia Services**	**Yes**	**Yes**	**Fully Homomorphic Encryption**	**Yes**	**Yes**

4 DECENTRALIZED PRIVACY-PRESERVING QOE SYSTEM EMPLOYING BLOCKCHAIN AND FHE

4.1 Problem Statement

QoE serves as a central metric for evaluating the services delivered by 6G networks, as these networks aim to support advanced user-centered applications. Improving QoE requires service providers to understand user satisfaction and expectations, which often involves collecting sensitive data. As discussed in Section 3, current privacy methods do not adequately permit service providers to perform computations directly on encrypted QoE data while ensuring data confidentiality and user control. The core challenge is a privacy-preserving method that allows verifiable computations on encrypted QoE data under user governance, within a decentralized infrastructure.

4.2 Design Goals

In alignment with the vision of 6G as more decentralized and user-centered than previous generations, with an emphasis on privacy protection and personalization [27, 103], we state three design goals that guide our architecture:

- **Privacy-preserving computation:** In the proposed architecture, the user QoE data must be kept encrypted at all stages: during storage, transmission, and computation. This approach prevents any party, including service providers, from accessing the raw data, thereby preserving user privacy.
- **Decentralized verification:** The system must operate without a central trusted authority. All stakeholders, including users, providers, and regulators, must be able to verify the correctness of each computation independently, supported by an underlying blockchain protocol.

- **User control:** Users must retain fine-grained control over which entities can run computations on their encrypted data, enabling personalized services while protecting privacy [80].

These goals motivate the architectural choices detailed in the following subsections.

4.3 System Model

Our system aims to provide end-to-end privacy for QoE data in emerging user-centric applications. It incorporates blockchain, FHE for computations on encrypted data, and user governance mechanisms to give individuals control over their data.

In the proposed QoE system, users can provide evaluations regarding a service to share their experiences with others while preserving their privacy. The system employs a decentralized infrastructure, meaning no single entity controls it, and users govern data sharing and actively participate in the evaluation process. We employed different QoE metrics in the evaluation section (Section 5). Additionally, we use the MOS to quantify user experience. It converts subjective perceptions into measurable data, allowing the evaluation of the service quality over time.

We argue that the system can be extended to include other QoE metrics since the proposed architecture can adapt to various QoE factors due to the decentralized operations of encrypted data. Moreover, 6G networks are expected to support a wide range of services with multiple requirements, necessitating an adaptable QoE system. Using FHE, service providers can compute and evaluate encrypted user data to improve their services without accessing raw user data. Similarly, regulatory bodies can monitor service provider compliance with mandated quality standards and SLA commitments by analyzing both the provider's encrypted performance data and encrypted user-reported QoE metrics - all while preserving the privacy of individual users and sensitive business information.

Our system model comprises the following key entities as illustrated in Figure 2:

- **Users:** End-users who assess and report their QoE. These users operate devices capable of collecting QoE metrics and interacting with the blockchain network. Users can grant permissions to service providers and regulators to access their encrypted data for specific purposes.
- **Service Providers:** Entities delivering services, such as video streaming or virtual reality applications, to users. They seek to improve service performance based on the QoE metrics collected.
- **Regulators:** Entities responsible for monitoring service quality and ensuring compliance with regulatory requirements. They can access aggregated QoE data to evaluate service performance and adherence to standards independently.
- **Validators (Network Nodes):** Blockchain network nodes responsible for participating in consensus, validating transactions, and executing the threshold decryption protocol. These nodes maintain the blockchain ledger and process smart contracts containing the FHE operations.
- **Blockchain Network:** A permissioned or public blockchain compatible with the EVM, offering a decentralized, transparent, and immutable platform for data storage and smart contract execution.
- **FHE Co-processor:** An off-chain component that extends the EVM environment by monitoring blockchain transactions for requests related to FHE operations. It manages ciphertext handles stored within smart contracts and performs the required FHE computations off-chain. These computations can optionally be accelerated using specialized hardware such as GPUs, FPGAs, or ASICs [37]. After completing the FHE operations, the co-processor submits the resulting ciphertexts back to the blockchain, ensuring they are publicly verifiable. This approach allows fully homomorphic encryption functionality to be supported within the existing EVM, without requiring changes to the underlying blockchain protocols.

The details of key management, noise handling, and decryption are provided in Section 4.4 and Section 4.5. The system interaction overview is as follows:

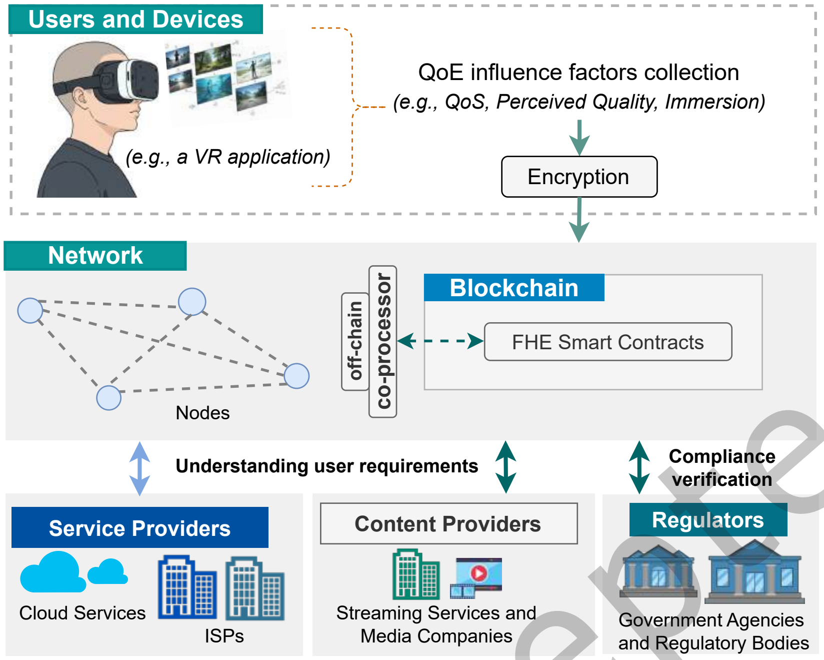

Fig. 2. An overview of key entities and interactions in our decentralized QoE system. Users encrypt their QoE metrics locally and submit them to the blockchain network. Service providers can perform computations on encrypted data through FHE-enabled smart contracts. Validators maintain the network and participate in threshold decryption when authorized. Regulators can access aggregate encrypted data to verify compliance. All interactions are governed by smart contracts with user-controlled access lists.

(1) QoE Data Collection and Encryption: Users' devices measure QoE Influence Factors (IFs) locally. This data is encrypted on the user's device using the global FHE public key (publicly available on the network).
(2) QoE Data Submission: Encrypted QoE data, along with zero-knowledge proofs, is submitted as a transaction to the blockchain network. Further details are provided in Section 4.5.
(3) Smart Contract Processing: Smart contracts on the blockchain receive and process the encrypted QoE data. These contracts maintain aggregate metrics, enforce access control, and can trigger computations on the encrypted data via the FHE co-processor.
(4) Homomorphic Computations: Service providers and regulators, with user-granted permissions, can request the smart contract to perform computations on the aggregated encrypted QoE data using the FHE co-processor.
(5) Threshold Decryption: When authorized entities (such as service providers or regulators with the necessary permissions) request decrypted aggregate results, the smart contract triggers a threshold decryption process. This process involves validators, as described in Section 4.4 and Section 4.5.
(6) Data Access and Auditing: Authorized entities can access decrypted aggregate QoE. Users can audit access logs and permission changes recorded on the blockchain.

4.4 Key Management and Noise Handling

Our system utilizes the Zama Protocol introduced in 2.3 that implements FHE compatible with EVM blockchains. The protocol employs a global FHE public key for encrypting all inputs and private states. This public key is openly available in the network, allowing new users to join the system and contribute their encrypted data without any complex key distribution process.

The system uses a single global encryption key, but the corresponding private decryption key is split among validators using a threshold protocol due to a decentralized Key Management Service (KMS). The initial validators generate this decryption key collaboratively in the setup phase, and it's redistributed when the validator set changes. This setup ensures that no group of validators smaller than a set threshold can decrypt data independently, and no single party ever holds the complete key. For any decryption to occur, a minimum number of validators must agree, combining their key shares. This method improves security by preventing individual validators from independently accessing or decrypting sensitive information. A preliminary analysis of threats and the security model is provided in Section 4.6.

In FHE schemes, noise accumulates as operations are performed on encrypted data, potentially affecting the accuracy of results. Our implementation leverages the TFHE (Fast Fully Homomorphic Encryption over the Torus) scheme, which supports bootstrapping to manage this noise. As introduced in Section 2.3, Bootstrapping refreshes the ciphertext, reducing noise and allowing for further computations. Optimizations in bootstrapping techniques have enabled these operations to be performed in milliseconds [20], making them practical for real-world applications [19, 29].

The frequency of noise management in *TFHE-rs* depends on several factors, including the encryption parameters, operation types, and desired correctness guarantees. For *shortint* operations, a "degree" attribute tracks the maximum encrypted value, triggering carry buffer emptying when a threshold is reached. Smart operations in *TFHE-rs* automatically handle noise by performing bootstrapping when necessary.

4.5 FHE-Enabled QoE Smart Contract Overview

The smart contract handles QoE evaluations using FHE. This contract leverages the fhEVM library to perform computations on encrypted data, ensuring that individual user inputs remain confidential while allowing for computation and verification. Figure 3 shows the layered architecture of the proposed system regarding the user interaction and the FHE operations. FHE operations are executed on a dedicated co-processor, an extension of the EVM that includes pre-compiled contracts with FHE operations (i.e., fhEVM). Therefore, while the network executes the logic of the contract in a decentralized manner, the cryptographic operations themselves are coordinated by an off-chain service as illustrated in Figure 2.

To support this architecture, a decentralized KMS is used to split and distribute the decryption keys, ensuring that no single party holds the complete decryption key. Additionally, a Zero-Knowledge Proof of Knowledge (ZKPoK) mechanism is employed behind the scenes to ensure that the operations are correctly performed and that the prover has knowledge of the plain-text values without revealing them, preventing tampering or misuse by unauthorized parties.

Users submit encrypted QoE metrics as inputs and zero-knowledge proofs, ensuring data validity while maintaining confidentiality. The contract maintains encrypted running totals for each metric, enabling ongoing analysis without exposing individual user data. The FHE co-processor enables computations directly on encrypted data while the smart contract updates the blockchain state. The blockchain infrastructure creates a permanent, tamper-proof record of all QoE submissions and computations. Since service providers can independently verify and perform calculations directly on encrypted data using FHE smart contracts, the system operates without relying on any central authority or trusted intermediary. Each computation and verification is transparently recorded on the blockchain while maintaining the privacy of the underlying user data.

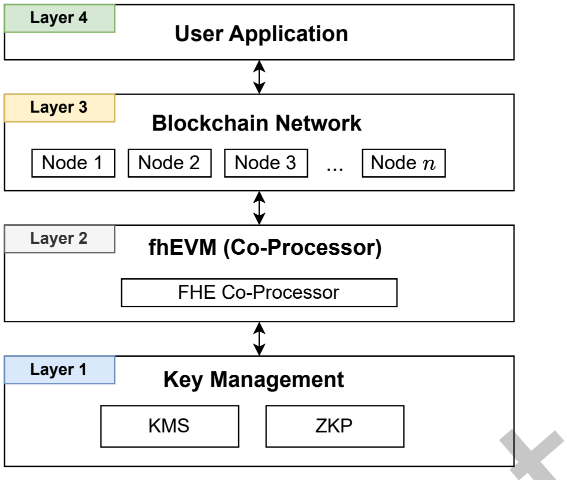

Fig. 3. Layer Architecture of a decentralized QoE system employing blockchain and FHE operations.

The contract incorporates an Access Control List (ACL) mechanism to enable user control. It allows users to manage permissions for their encrypted data dynamically. Users can set and modify access rights, specifying which addresses (such as those belonging to service providers or telecommunications companies) are authorized to perform homomorphic computations on the user's data. The system allows users to grant or revoke these permissions at any time. If a user revokes access, it prevents previously authorized parties from conducting further operations on that user's data.

The Algorithm 1 represents the contract state update for QoE metrics using homomorphic operations. While FHE enables various types of operations on encrypted data (including multiplication, subtraction, comparisons, and bitwise operations), this algorithm demonstrates the state update process using the addition operation as an example. The contract maintains encrypted data entries and totals for each QoE metric, enabling service providers to analyze and improve services through computations on encrypted data. Access to these computations must be explicitly granted through the ACL using the *TFHE.allow* method.

The Algorithm 2 represents the data decryption process in the FHE smart contract. The contract requests decryption for encrypted data fields, triggering a callback function when the decryption is complete. When the *requestDecryption* method is invoked, the threshold decryption protocol is executed, ensuring that a minimum number of validators participate in the decryption process.

A note on user and company privacy: The evaluations from users are encrypted on their devices before being submitted. This ensures that the raw scores do not leave the user's device in an unencrypted form. This means that even during processing, individual ratings remain confidential (i.e., encrypted).

Companies can also implement smart contracts to enable data sharing with specific stakeholders, such as regulators, for compliance verification. This can be achieved using ACLs to manage permissions. Importantly, both users and companies can monitor the operations conducted by others, ensuring transparency.

Algorithm 1 Contract State Update for QoE Metrics Using Homomorphic ”Add” Operation

```
Input: Encrypted QoE metrics: $Enc[] = \{Enc01, Enc02, \ldots, EncN\}$
Output: Updated totals: $Tot[] = \{Tot01, Tot02, \ldots, TotN\}$
procedure AddData($Enc[]$)
    Update Entries:
    Add encrypted values $Enc[] = \{Enc01, Enc02, \ldots, EncN\}$ to dataEntries
    Update Totals:
    for each $Enc_i$ in $Enc[]$ do
        $Tot_i \leftarrow$ TFHE.add($Tot_i$, TFHE.asEuint32($Enc_i.value, Enc_i.proof$))
    end for
    Allow Operations on Totals:
    for each $Tot_i$ in $Tot[]$ do
        Call TFHE.allow($Tot_i$, address(this))
    end for
end procedure
      return Updated totals: $Tot[]$
```

Algorithm 2 Data Decryption in FHE Smart Contract.

```
Input: Encrypted data fields: $Enc[] = \{Enc_1, Enc_2, \ldots, Enc_n\}$
Output: Decrypted data fields: $Dec[] = \{Dec_1, Dec_2, \ldots, Dec_n\}$
Initialize: array $Dec[]$
for each entry in dataEntries do
    Create $cts[]$ (ciphertext array) for all encrypted fields
    for each $Enc_i$ in entry do
        Add $Enc_i$ to $cts[]$
    end for
    Request Decryption:
    Call Gateway.requestDecryption(cts[], callbackSelector)
end for
Decryption Callback:
for each decrypted $Dec_i$ in callback do
    Store $Dec_i$ in $Dec[]$
end for
      return $Dec[]$
```

4.6 Security Model and Threat Analysis

This section analyzes key security risks and mitigation strategies for the proposed privacy-preserving QoE system. The system assumes the security of underlying cryptographic primitives (FHE, ZKPoK, threshold MPC). It relies on threshold key management where at least $t + 1$ out of n validators remain honest. Security goals include end-to-end confidentiality, privacy-preserving computation, decentralized key management, data integrity, granular access control, ciphertext misuse prevention, and secure key rotation.

4.6.1 Security Assumptions and Objectives. The security analysis relies on the following assumptions:

- **Cryptographic Primitives:** Assumes security of TFHE, KMS of Zama protocol under the *TFHE-rs* implementation [98]. Also, adversaries are assumed to have bounded computational resources, insufficient to break strong cryptographic primitives such as FHE and threshold decryption.
- **Byzantine Fault Tolerance:** Assumes BFT-based consensus requiring at least $3f + 1$ validators with at most f faulty validators as proposed by Castro et al. [15].
- **Threshold Description:** Assumes at least a threshold $t + 1$ out of n KMS validators are honest and do not collude beyond the threshold.

The system is designed to achieve:

- **Privacy-Preserving Computation:** Ensure that user QoE data remains encrypted during storage and computation.
- **Integrity:** Once created, guarantee that QoE data is immutable.

We focus on five main threats: data privacy breaches, decryption key reuse, key rotation vulnerabilities, malicious smart contracts, and ciphertext misuse. For each threat, we outline the potential risks and technical mitigations.

4.6.2 Data Privacy Breaches.

- **Threat:** Unauthorized access to individual user QoE data.
- **Mitigation:** FHE ensures that all user data remains encrypted end-to-end, even during computations. Formally, for a plain-text message $m \in M$, encryption function Enc, and any function f:

$$Enc(m) \rightarrow c \tag{1}$$

$$f(Enc(m)) = Enc(f(m)) \tag{2}$$

  This property ensures that computations on encrypted data yield encrypted results, preserving privacy throughout the process. Additionally, decryption only occurs when specific decryption methods are called, and these operations require a threshold number of validators to participate in the decryption process. This multi-party computation approach protects against unauthorized access to individual user data, as no single entity can unilaterally decrypt the information.

4.6.3 Decryption Key Reuse.

- **Threat:** Once the decryption key is generated, malicious actors might attempt to reuse it for unauthorized decryptions.
- **Mitigation:** To avoid it, the system employs a threshold decryption protocol to share pieces of secret key $(s)'$ with Ephemeral partial keys ($[E]_i^{\langle t,Q\rangle}$) [98]. The process is as follows:
  1. **Key Sharing:** The secret key $s' \in 0, 1^L$ is secret shared modulo Q across n parties with a threshold $t < n$ as $[s']^{\langle t,Q\rangle}$. Each bit of $s'_j \in s'$ is shared separately.
  2. **Ciphertext Preparation:** For a ciphertext $c = (a, b) \in \mathbb{Z}_q^\ell \times \mathbb{Z}_q$, the parties first apply $c' \leftarrow$ Switch-n-Squash(c) to get $c' = (a', b') \in \mathbb{Z}_Q^L \times \mathbb{Z}_Q$.
  3. **Noise Generation:** Parties jointly generate a secret shared noise term $[E]^{\langle t,Q\rangle}$.
  4. **Partial Decryption:** Each party P_i computes a partial decryption:

$$PDec_i(c') = b' - \langle a', [s']_i^{\langle t,Q\rangle}\rangle + [E]_i^{\langle t,Q\rangle} = [e' + E + \Delta' \cdot m]_i^{\langle t,Q\rangle} \tag{3}$$

     Where:
     - e' is the error present in the encryption
     - $\Delta' = Q/p$, where p is the plain-text modulus

  (5) **Reconstruction:** Combining more than t valid $PDec_i(c')$ values allows reconstruction of the message m.

  This process ensures:
  - Each decryption operation requires a new initiation of the threshold protocol.
  - Ephemeral partial keys ($[E]_i^{\langle t,Q\rangle}$) are generated for each decryption operation and cannot be reused.
- **Security Guarantee:** This threshold decryption scheme is secure against up to t corrupted parties, as long as $t < n/4$ in the asynchronous network setting, or $t < n/3$ in the synchronous setting with timeouts.

4.6.4 *Key Rotation.*

- **Threat:** Over time, the security of cryptographic keys may degrade due to advances in cryptanalysis or the accumulation of partial information from multiple decryption operations.
- **Mitigation:** The system implements a key rotation mechanism based on the dynamic-committee proactive secret sharing (DPSS) protocol [98]. The process is as follows:
  1. **Committee Change:** When transitioning from one set of validators $S = \{P_1, \dots, P_n\}$ to a new set $S' = \{P'_1, \dots, P'_m\}$:
  2. **Share Distribution:** Each party $P_i \in S$ shares their key share $[s]_i^{\langle t,Q\rangle}$ to the parties in $P_j \in S'$ using a Verifiable Secret Sharing (VSS) scheme.
  3. **New Key Share Construction:** By applying the recombination vector for the sharing in S to the shares of the shares, parties in S' construct a sharing of the original secret s.
- **Security Guarantee:** This DPSS protocol ensures that:
  - The secret key s is securely transferred to the new committee without ever being fully reconstructed.
  - The new shares are independent of the old shares, preventing the accumulation of information over multiple rotations.

4.6.5 *Malicious Smart Contracts.*

- **Threat:** Unauthorized extraction or manipulation of sensitive encrypted QoE data by malicious smart contracts.
- **Mitigation:**
  1. **Handle-based Access:** Smart contracts interact with data via handles: $h_c := \text{Keccak256}(c)$
  2. **Honest Ciphertext Tracking:** System maintains $\mu : h_c \mapsto (c, S)$, where S is the set of valid call stack depths
  3. **Threshold Decryption:** Requires $t + 1$ out of n validators: $m = \text{Combine}(\{\text{PartialDec}_i(c)\}_{i \in T})$
- **Security Guarantee:** Prevents direct ciphertext access, ensures honest ciphertext usage, and requires multi-validator consensus for decryption.

4.6.6 *Ciphertext Misuse in Unintended Contexts.*

- **Threat:** Unauthorized reuse of ciphertexts in contexts other than their intended purpose, potentially leading to security vulnerabilities or privacy breaches.
- **Mitigation:** The system employs a robust certification mechanism for ciphertexts:
  - Users must submit a certified ciphertext consisting of the following:
    1. The ciphertext itself: $c = Enc_{pk}(m)$
    2. An associated valid zero-knowledge proof of plain-text knowledge (ZKPoK): π
  - The ZKPoK ensures:
    1. Ciphertext well-formedness
    2. User knowledge of the underlying plain-text message m
    3. Context-specific binding of the ciphertext

- Implementation via a signature-proof-of-knowledge scheme [14, 16]:

$$\pi = SPK\{(m) : c = Enc_{pk}(m) \wedge \text{Statement}(m)\}(\text{context}) \tag{4}$$

  Where:
  - SPK denotes a signature-proof-of-knowledge
  - $\text{Statement}(m)$ is a predicate about the plain-text
  - context includes user address, smart contract address, and function selector

5 PROOF-OF-CONCEPT IMPLEMENTATION AND EVALUATION

As presented in Figure 4, the system consists of four components in the user application: the QoE metrics collection (influence factors), data encoding, privacy protection, and transaction creation module. The user application allows users to interact with the system, submit evaluations, and view users' satisfaction with different services. Our system evaluates the services' QoE by calculating the user's individual MOS computed locally on the user's device, considering various QoE factors specific to the service type (e.g., video quality, loading time). While MOS can also be estimated using machine learning models that consider QoS parameters and factors like affective computing (e.g., facial expressions) [1, 10], the details of MOS calculation and estimation are beyond the scope of this study. We focus on scenarios where the user application has already calculated the MOS based on the user's experience.

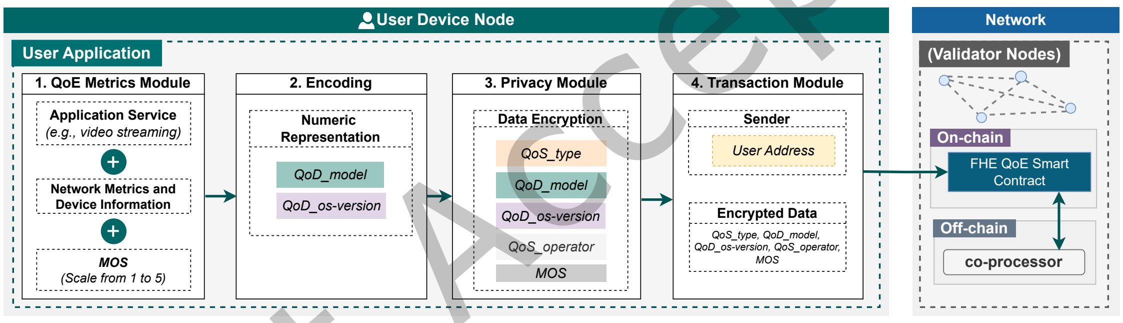

Fig. 4. Proposed QoE system flow involves collecting QoE influence factors in the user application, which are then processed using a privacy-preserving mechanism employing FHE. Additionally, the system includes a transaction module to enable users to submit their evaluations.

To evaluate the system, we utilized the PoQeMoN dataset [56]. This dataset was created using a custom Android application designed to evaluate YouTube users' QoE across different mobile networks, including UMTS, HSPA, and LTE. The application, incorporating a VLC media player, collects a wide range of QoE IFs from users' devices during real-world mobile video streaming sessions. The dataset contains 1543 samples from 4 different network operators and 20 unique device models. It comprises 23 columns capturing various QoE influence factors and the corresponding MOS.

Table 2 shows the evaluation setup, including the technologies and libraries used for the proof-of-concept evaluation. The Zama Protocol was employed for privacy-preserving computations using the fhEVM library, an EVM implementation that supports fully homomorphic encryption [98]. The system was evaluated using the PoQeMoN dataset to demonstrate the feasibility of the proposed approach in a real-world scenario. For network validation, we utilized both a Zama local development node (BFT consensus) and the Sepolia testnet

(Proof-of-Stake consensus). Additionally, load testing was performed using Apache JMeter to generate concurrent transaction workloads to establish a baseline under a centralized server architecture.

Table 2. Proof-of-Concept Evaluation Configuration.

Configuration	**Feature / Tool Employed**
Dataset	PoQeMoN [56]
Privacy Scheme	Zama Protocol (fhEVM library) [98]; Fast FHE over the Torus [20] in Rust (TFHE-rs) [100]
Environment	Linux Ubuntu 22.04; Intel Core i7-13650HX (20 cores, 2.80GHz); 16GB RAM
Local Network	Zama local development node (BFT consensus)
Public Network	Sepolia testnet (Proof-of-Stake consensus)
Load Test	Apache JMeter [2]
Software Availability	All source code is available on GitHub [35]

In our system, the provider can perform a variety of analytics on encrypted data to generate reports on QoE without accessing individual raw user data. For instance, the provider can calculate the average MOS score across different *QoS_operator* categories without decrypting individual scores. Additionally, the system allows the provider to sum buffering times for different video types, determine the maximum frame drop rate across device models, and analyze the age distribution of users within various MOS score ranges. Furthermore, the system enables the provider to determine the median video quality score for various device types and compute the ratio of satisfied users for each network operator, all without decrypting the underlying data. These operations enable the provider to optimize service delivery and enhance user satisfaction while ensuring the confidentiality of user information.

The sensitive information in this dataset includes individual user demographics (such as age and gender), detailed device specifications, network usage data, and subjective quality ratings (MOS scores), all of which could reveal personal preferences and behavior patterns if not properly protected. We employ a decentralized infrastructure that ensures all stakeholders, including users and service providers, can independently verify each computation due to the transparency of encrypted data and operations, along with the deterministic nature of FHE [98]. The system enables direct comparison of operators' performance through objective metrics. Since these metrics are public, operators must maintain high service standards to retain customers and attract new ones. Poor service quality is visible to users, regulators, and (with user permission) other service providers through access control lists, enforcing accountability.

5.1 Encryption Evaluation

We utilized sensitive columns (see Table 3) in the dataset for our encryption evaluation to represent stages 1, 2, and 3 of Figure 4. In the encryption process, the protocol uses a global FHE key as a public key under which all inputs and private states are encrypted. The use of a single public key allows composability between smart contracts and multi-user applications [98]. The decryption process uses a distributed threshold protocol where multiple validators participate collectively to decrypt the data, as introduced in Section 2.3. Each validator holds a share of the secret decryption key, and no single party knows the full key. When decryption is needed, validators execute an interactive protocol using their key shares to jointly decrypt the ciphertext. This approach ensures that a minimum number of validators above a certain threshold must cooperate to perform decryption, enhancing security and preventing any single party from accessing the plain-text data independently.

We utilized all 1543 samples in the dataset. Table 3 presents the selected columns and their details. Note that our proposed system can be adapted to handle additional columns based on the specific requirements of the service provider and the nature of the data being processed. The system allows for the inclusion of various QoE influence factors and the evaluation of different service types. Additionally, we are not sending users' personal information to the blockchain due to privacy regulations such as the General Data Protection Regulation (GDPR) that impose right-to-be-forgotten principles [8]. We analyzed five key metrics - QoS Type, QoD Model, QoD OS Version, QoS Operator, and MOS. We encrypted the items using the Zama Protocol employing its JavaScript library (fhevmjs) [99] with 8 bits due to the range of values.

Table 3. Sensitive Columns and Details for Encryption Evaluation.

Dataset Column	Value Range	Related to	Details
QoS_type	1-5	Service Operator	The type of service can reveal information about a user's mobile service plan
QoD_model	1-15	Device	The device model can potentially be used to identify individual users
QoS_operator	1-4	Service Operator	The mobile service operator information
QoD_os-version	1-18	Device	Operating system version
MOS	1-5	User	User subjective quality rating

Using the sensitive QoE impact factors listed in Table 3, we evaluated the performance of local encryption under concurrent request loads of 10, 50, and 100. To conduct the tests, we used Apache JMeter to simulate client-server interactions via an API endpoint that performs encryption operations. Figure 5 highlights scalability challenges as request concurrency increases. Encryption latency increased under higher concurrency, changing from approximately 0.1 seconds at 10 requests to 1.1 seconds at 100 requests due to server-side processing demands. Similarly, CPU utilization scaled proportionally to request volume. Alternatives for encryption and FHE operations performance, such as hardware acceleration schemes [37], are out of the scope of this study.

5.2 Homomorphic Operations

We established a baseline by evaluating addition, subtraction, and multiplication operations within a centralized server architecture using the *TFHE-rs* library [100]. Similarly to encryption, we used Apache JMeter to simulate client-server interactions via an API endpoint implemented in Rust, which handles homomorphic computations on encrypted data. We measured latency for single-user requests and throughput under concurrent multi-user load conditions. Figure 6 presents the performance of homomorphic encryption operations in a centralized server setup. The chart on the left shows the latency distribution for basic arithmetic operations on MOS data under single-user conditions. Under single-user conditions, the observed latencies remained in the millisecond range, consistent with previously reported results in [100]. The chart on the right shows how latency changes as the number of concurrent users increases, providing a view of system scalability.

Although centralized computation achieves processing times on the order of milliseconds, it introduces a single point of failure and centralizes control over all operations. In contrast, the proposed architecture prioritizes decentralization and privacy-preserving computation on encrypted QoE data, while still ensuring public verifiability due to blockchain transparency.

5.3 Smart Contract Evaluation

5.3.1 Deployment and Transaction Cost. We evaluated the cost of deploying smart contracts that execute arithmetic operations on encrypted data, as listed in Table 3. These operations include addition (Add), subtraction (Sub), and multiplication (Mult). The deployment cost was measured in Ethereum (ETH) on the Sepolia public

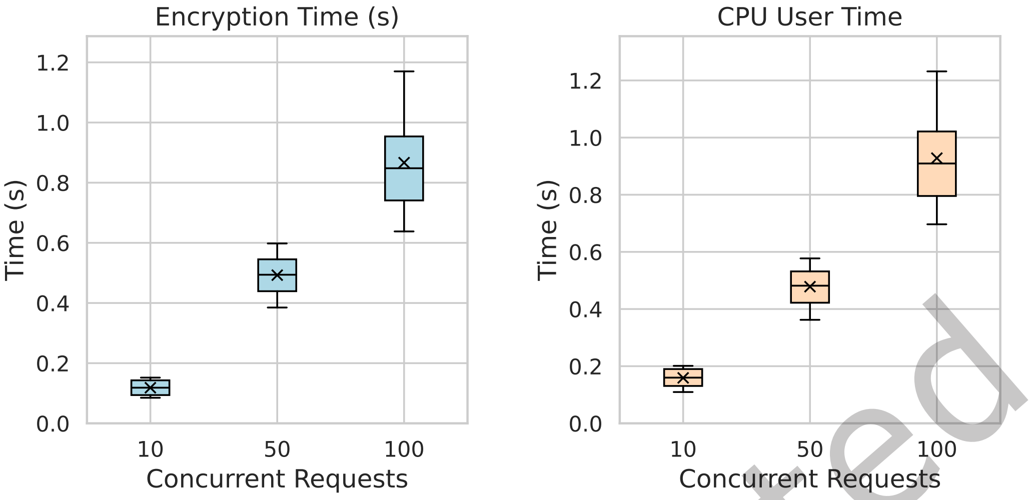

Fig. 5. Local encryption evaluation metrics, including encryption time and CPU usage under concurrent user loads for the transaction items listed in Table 3. The measurements focus on stages 3 and 4 of the encryption process (see Figure 4), showing how system performance and resource consumption change with increasing user load.

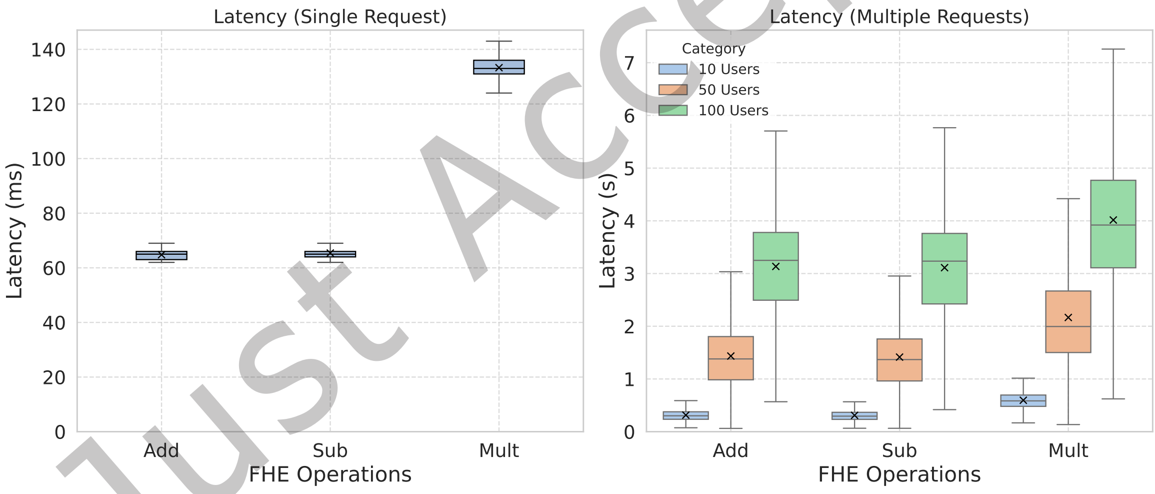

Fig. 6. Performance analysis of centralized homomorphic encryption operations. (Left) Latency distribution under single-user conditions. (Right) varying user loads.

test network. The deployment was conducted using a JavaScript client using the web3.js library to send the deployment transaction. Figure 7 shows the deployment cost for each operation. The compiled contract's bytecode is 3595 bytes for the Add operation, 3028 bytes for Sub, and 3528 bytes for Mult. The deployment cost was similar across operations, ranging from 0.000794 ETH to 0.000907 ETH. Since these contracts perform basic operations and are rarely redeployed. All the smart contract code is available in the GitHub repository [35]

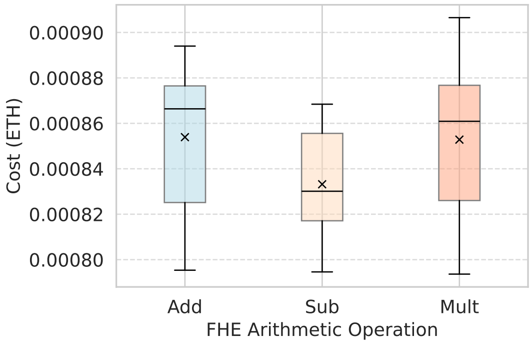

Fig. 7. Deployment cost (in ETH) for smart contracts implementing arithmetic operations—specifically Addition, Subtraction, and Multiplication—for all items listed in Table 3 on the Sepolia public testnet.

Similarly, Table 4 shows the transaction costs on the Sepolia testnet for the smart contracts containing arithmetic operations. These transactions, each with a size of 0.16 KB, had a mean confirmation time of approximately 11 seconds, as detailed in the table.

Table 4. Evaluation of transaction costs on the Sepolia public testnet, with a focus on processing operations over encrypted data.

FHE Operation on QoE Metric	**Transaction Size (KB)**	**Transaction Confirmation Time (s)**				**Transaction Cost (ETH)**			
		mean	std	min	max	mean	std	min	max
Add	0.16	11.52	0.85	9.73	12.90	0.000929	0.0	0.000929	0.000929
Mult	0.16	11.58	0.74	10.83	12.86	0.000708	0.0	0.000708	0.000708
Sub	0.16	11.67	0.84	9.70	12.95	0.000709	0.0	0.000708	0.000710

5.3.2 Transaction Confirmation Time. Figure 8 shows two performance metrics: the transaction confirmation time by a consensus node in a local network, and the confirmation time on the Sepolia public blockchain. These metrics correspond to the user submissions of all encrypted data presented in Table 3 and the subsequent transaction confirmation by the network, which completes the FHE operations.

In the local network, the average transaction confirmation time was 8.78 seconds, with a standard deviation of 0.61 seconds and a range of 7.11 to 10.22 seconds. This evaluation used the default configuration of the BFT consensus protocol, with block creation set at 5-second intervals. This narrow range indicates consistent processing performance for the smart contract described in Section 4.5. In comparison, the Sepolia public testnet showed a higher average confirmation time of 11.52 seconds, with a standard deviation of 0.84 seconds and a range from 9.73 to 12.97 seconds. The slower performance on Sepolia reflects the added overhead of its public consensus process. However, the FHE operations do not introduce enough overhead to impact the block creation process.

It's important to note that the complexities of FHE operations can increase significantly with larger datasets and higher numbers of user submissions. Additionally, communication delays within the network can substantially affect the overall system performance, potentially increasing transaction confirmation times and influencing the coordination of FHE computations across nodes.

A note on high transaction rate: Since the focus of this study is on the privacy perspective employing FHE operations, this study does not evaluate system performance under high QoE transaction submission rates from users. Scalability remains a challenge, particularly when handling large transaction volumes. The potential impact of high QoE data submission rates on blockchain scalability is not examined in this work. In future work, we plan to address this limitation by evaluating system performance under high transaction rates and exploring scalability solutions, such as layer-2 protocols and alternative consensus mechanisms.

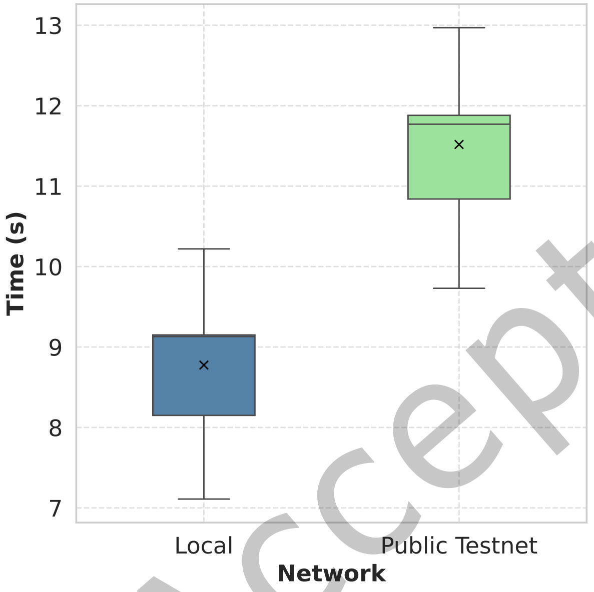

Fig. 8. Comparison of transaction confirmation time between local execution and the Sepolia public blockchain testnet.

5.4 Comparative Analysis with Existing Solutions

To position our decentralized and privacy-preserving QoE system, we provide a comparative analysis with existing approaches. While comprehensive quantitative benchmarking of all potential solutions is beyond the scope of this proof-of-concept study, we present a qualitative comparison highlighting key features and conceptual trade-offs. We also plan to conduct detailed quantitative benchmarks in future work to further evaluate the performance of the proposed architecture in relation to these existing methods. Table 5 presents a feature-based comparison of various QoE approaches, including our proposed architecture, centralized cloud-based QoE, Federated Learning for QoE, and Differential Privacy applied to QoE data.

Centralized Cloud-Based QoE systems prioritize performance but inherently expose user data to the service provider, leading to significant privacy risks. Federated Learning offers improved privacy by enabling distributed model training, but trade-offs remain. While FL reduces data centralization, it introduces communication overhead [5, 44] and still presents risks of information leakage through aggregated model updates. Differential Privacy provides a quantifiable privacy measure, but the necessity to add noise to achieve privacy impacts data utility and QoE accuracy [31, 57].

In contrast, the proposed architecture combines blockchain for decentralization with FHE for computation on encrypted data. This combination is designed to provide end-to-end data confidentiality, verifiable transparency

Table 5. Comparative Analysis of Different QoE Management Approaches.

	Existing Works			
Feature	**Centralized Cloud-Based QoE** **[3], [54]**	**Federated Learning for QoE** **[70],[85],[73],[58],[68] [63],[21],[59],[72]**	**Differential Privacy for QoE** **[48], [31], [101]**	**Proposed Blockchain-Based QoE Architecture Using FHE**
Privacy Preservation Level	Low (Data exposed to central provider)	Improved (Local training, but aggregation risks)	Moderate (Privacy for aggregate data, utility trade-off)	High (End-to-End Encryption, Computation on Encrypted Data)
Decentralization	Centralized	Federated	Centralized (Data collection/analysis often centralized)	Decentralized (Blockchain-based)
Transparency and Auditability	Limited (Provider-controlled)	Limited (Aggregation process opaque)	Limited (DP mechanism can be complex)	High (Blockchain immutability and transparency)
User Control and Governance	Low (Users have limited control over data usage)	Low to Moderate (Limited user control in typical FL setups)	Low (Primarily algorithmic privacy)	High (Smart Contract ACLs, User-Managed Permissions)
Computation on Encrypted Data	No	No	No	Yes (Fully Homomorphic Encryption)
Trust Model	High Trust in Central Provider	Trust in Central Aggregator	Trust in DP Mechanism and Data Curator	Reduced Trust (Trust in Cryptography and Decentralized Network)
Computational Overhead	Low	Moderate (Distributed training, communication costs)	Low (Noise addition in DP)	High (FHE Operations, Bootstrapping)
Deployment Complexity	Low	Moderate (Federated setup, coordination)	Low to Moderate (DP implementation and parameter tuning)	Moderate to High (Blockchain integration, FHE Smart Contracts)
Primary Advantage	Simplicity, Performance Optimization	Privacy-Aware Model Training, Reduced Data Centralization	Privacy-Preserving Aggregate Statistics	End-to-End Data Confidentiality, User Empowerment, Verifiable Transparency
Key Trade-off	Privacy Risks [86]	Potential Information Leakage and Communication Overhead [5], [44]	Utility Loss and Privacy-Utility Trade-off [31], [57]	Computational Overhead and Latency [40], [12]

through blockchain, and granular user control via smart contracts — features not simultaneously offered by the other approaches. However, this design also introduces trade-offs. Both homomorphic encryption operations and blockchain consensus mechanisms increase processing latency. Unlike conventional encryption methods, FHE allows computations to be performed directly on encrypted data at all stages, preserving confidentiality throughout the process. To address performance challenges, future work will explore optimization mechanisms to improve transaction processing rates while using FHE.

6 DISCUSSION

6.1 Performance Implications and Computational Overhead of FHE

FHE provides privacy guarantees for user data, as it is kept encrypted during processing. Some improvements have been made to make FHE more practical [37, 96]. However, even with these performance improvements, FHE remains slower than computation on unencrypted data. This is due to the inherently high computational cost of operating on encrypted inputs, the large size of ciphertexts containing sensitive user data, and the need for frequent bootstrapping steps to control noise accumulation [37]. Its performance characteristics are complex, as the overhead scales with both the number of encrypted inputs and the complexity of the operations performed.

In our study, our proof-of-concept evaluates a scenario with multiple concurrent users submitting encrypted QoE metrics derived from the real-world video streaming dataset [56]. These metrics, detailed in Table 3, include sensitive information like user MOS, device OS versions, and network operator details. In the setup configuration shown in Table 2, the results in Figure 5 show a performance degradation as the system load increases; the average encryption latency grew from milliseconds under a light load to over a second for 100 concurrent requests, with a corresponding rise in CPU utilization. The performance of the homomorphic operations themselves, as detailed in Figure 6, also highlights this scalability issue. Even simple arithmetic functions on encrypted data exhibit millisecond-range latencies that degrade under concurrent user loads.

To mitigate this latency, our architecture utilizes an off-chain FHE co-processor, as presented in Figure 4. This design decouples heavy cryptographic tasks from the blockchain, allowing them to be handled by dedicated hardware accelerators without altering the underlying blockchain protocol. This design provides the architectural flexibility to enhance performance for demanding 6G networks, as the co-processor can be optimized independently of the core protocol.

6.2 System applicability and adoption in 6G networks

As 6G networks transition toward decentralized architectures, they are expected to support a diverse range of services, each with distinct performance and requirements. Our system's flexibility allows for incorporating QoE impact factors beyond those used in the current proof-of-concept (QoS type, device model, OS version, and network operator). It can include factors relevant to emerging 6G services.

The system's use of fully homomorphic encryption ensures user privacy protection as QoE data complexity increases in user-centric and context-aware 6G networks. Centralized FHE computation achieves processing times in the millisecond range but creates a single point of failure and concentrates control over all operations. In contrast, our decentralized infrastructure eliminates the need for a centralized authority to manage QoE, enabling decentralized operations across different network nodes and service providers. The system supports verifiable computation: each node independently validates computations based on the transparency of encrypted data and the deterministic behavior of FHE operations.

Additionally, service providers can create smart contracts to analyze aggregate QoE data and automatically adjust their offerings. These contracts can execute computations on encrypted data, for example, adjusting network resource allocation or triggering service improvements using certain thresholds. Due to the decentralized nature of the system, regulators and other stakeholders can independently monitor service quality and verify compliance without relying on a central authority or trusted intermediary while maintaining the privacy of both user data and sensitive business information through encrypted computations.

However, 6G will introduce potential scalability challenges related to blockchain technology, particularly in processing the high transaction rates [71]. As 6G networks are designed to support massive connectivity and real-time applications, the volume of QoE evaluations and related transactions could be substantial. Future research should explore scalability solutions for the blockchain component, such as sharding, layer-2 solutions, or alternative consensus mechanisms, as proposed by Ni et al. [71], to ensure the system can handle the demands of high-transaction environments expected in 6G networks without compromising the benefits of decentralization and transparency.

6.3 How does the proposed system address the research questions?

Our research aimed to address the following three key questions regarding the implementation of a decentralized, privacy-preserving QoE system for 6G networks:

(1) **How does transparency in a decentralized infrastructure impact user trust and enable service providers to understand users' needs?**
The proposed system uses a decentralized infrastructure to ensure transparency and enable trust in computation. Users can confirm that their metrics are included in aggregate calculations without exposing

sensitive data. The system supports verifiable computation: each node independently validates computations based on the transparency of encrypted data and the deterministic behavior of FHE operations.

Furthermore, the system enables service providers to share their service metrics with users and regulators in a privacy-preserving manner using ACLs. This feature allows users to verify that the services they receive correspond to what they originally subscribed to or requested without compromising the privacy of other users or sensitive business information. For instance, service providers can demonstrate compliance with quality standards or SLAs by sharing encrypted performance data that can be compared against user-reported QoE metrics.

Service providers may be reluctant to share sensitive data with competitors for competitive reasons. Using ACLs and encrypted data ensures that service providers can selectively share performance metrics with users and regulators without exposing private information to other companies. Protecting proprietary information encourages more open sharing of quality metrics, as providers can prove their service quality without risking their competitive advantage. Due to homomorphic encryption, regulators can perform necessary audits without accessing individual user data or proprietary company information. It enables a more trusting environment where service quality can be verified while respecting all parties' privacy and business interests. Moreover, the system's decentralized nature prevents any single entity from controlling the process.

(2) **How can end users govern data sharing and computations on their encrypted QoE data with authorized parties (such as service providers and regulators), enabling these parties to perform operations on the data while protecting both user privacy and business confidentiality?**

Users can dynamically manage access and computations to their encrypted QoE data using ACL in the smart contract functions. Our system allows users to control and audit who can access their data, for how long, and for what purposes, while allowing for computations and analysis. It allows users to grant specific permissions for individual QoE metrics to different addresses, such as service providers or auditors.

The blockchain provides a built-in auditing capability. Since permanent ACL changes are recorded on the blockchain, users can review the access history of their data, seeing when and to whom access was granted or revoked. This transparency allows users to monitor and verify how their data is used over time. Additionally, consumers and regulators can make encrypted computations using metrics provided by service providers to verify compliance, comparing actual performance against contracted service levels without compromising sensitive business information or individual user privacy. This end-to-end verifiability ensures transparency while maintaining the confidentiality required by all stakeholders.

Our system employs fully homomorphic encryption to allow users to contribute their QoE metrics in an encrypted form. This approach enables users to share their experiences without exposing personal data. Service providers can identify areas for improvement and optimize their services based on user evaluation. The smart contract's ability to handle computations on encrypted data means that operations can be performed without decrypting individual user data, maintaining user privacy.

For regulators, the blockchain provides an immutable record of QoE metrics. A key feature of our system is its ability to adapt to changes in the verification of service-level agreements (SLAs) through smart contracts. These contracts enable comparison between actual service performance and contracted service levels through encrypted computations. For instance, providers can demonstrate compliance for network performance metrics like bandwidth, latency, and uptime by performing FHE operations that compare encrypted measurements against contracted thresholds. This approach ensures regulatory verification while maintaining the confidentiality of the underlying data throughout the process. While our current implementation does not include a specific smart contract for SLA verification, the system's architecture using FHE is well-suited for such extensions. As future work, we plan to investigate specific smart contract implementations for SLAs that incorporate encrypted verification.

By encrypting QoE metrics and excluding personal data from blockchain transactions, we address the conflict between GDPR's right to be forgotten and blockchain technology's immutable ledger. This approach allows leveraging blockchain technology's benefits for transparent and tamper-proof QoE data collection while respecting and protecting user privacy. Even though the data remains encrypted, companies and service providers can still make computations to improve their services using smart contracts. Due to multi-user participation, the system generates a dataset that provides an understanding of QoE, leading to service improvements.

7 CONCLUSION

This work presented a novel decentralized and privacy-preserving system for managing the quality of experience in 6G networks. The proposed study leverages blockchain technology and fully homomorphic encryption compatible with any EVM-based blockchains to address the challenges of user privacy in QoE management. Our approach enables users to actively contribute their QoE evaluations while maintaining the privacy of their sensitive data. By employing FHE, the system allows service providers to perform computations on encrypted user data to improve service quality without compromising individual user privacy. Blockchain's data integrity and the deterministic nature of FHE operations allow both users and regulators to independently verify computations and compliance. This verification occurs without compromising privacy or revealing providers' confidential business data, as all computations remain encrypted.

We provided a security assumption and threat analysis, examining key security risks, including data privacy breaches, decryption key reuse, key rotation vulnerabilities, malicious smart contracts, and ciphertext misuse. We analyzed potential risks for each threat, described our technical mitigation strategies, and explained the resulting security guarantees. Additionally, we evaluated our proof-of-concept implementation using PoQeMoN, a real-world dataset of mobile video streaming sessions. Additionally, we developed the smart contracts employing the *fhEVM* library, which uses a Rust implementation of Fast, Fully Homomorphic Encryption over the Torus. Our evaluation indicated consistent performance in both encryption times and transaction processing, indicating the system's potential for practical application in existing blockchains with increased delay of milliseconds in smart contract operations. Our proposed system is flexible to different QoE metrics and service types, allowing it to adapt to the wide-ranging and evolving requirements of future 6G applications.

In this work, we focused on the privacy aspect of QoE data. Future work will address three primary areas. First, we will investigate blockchain scalability to handle the large volume of QoE data expected in 6G networks. This will involve improving the blockchain infrastructure and consensus mechanisms to ensure the system can operate under high transaction loads in real-world conditions. Second, quantitative benchmarking will be conducted to assess the performance of the proposed architecture and compare it against existing privacy-preserving solutions. A formal security analysis will also be performed to evaluate the security properties offered by the combined use of blockchain and FHE. Third, we plan to include automated SLA management through the use of confidential smart contracts combined with FHE. This approach will enable the automatic monitoring of compliance, trigger actions based on encrypted QoE metrics, and support dynamic adjustments to SLAs, all while preserving data privacy.

ACKNOWLEDGMENTS

This work was supported by Ericsson Telecomunicações Ltda, and by the Sao Paulo Research Foundation, grant 2021/00199-8, CPE SMARTNESS, and grant 2023/15919-1.

REFERENCES

[1] Lamine Amour, Mohamed-Ikbel Boulabiar, Sami Souihi, and A. Mellouk. 2018. An improved QoE estimation method based on QoS and affective computing. *2018 International Symposium on Programming and Systems (ISPS)* (2018), 1–6. https://doi.org/10.1109/ISPS.2018.

8379009

[2] Apache Software Foundation. 2025. Apache JMeter. https://github.com/apache/jmeter. Accessed: 2025-04-16.

[3] Jesus Arellano-Uson, Eduardo Magaña, Daniel Morato, and Mikel Izal. 2024. Survey on Quality of Experience Evaluation for Cloud-Based Interactive Applications. *Applied Sciences* 14, 5 (Feb. 2024), 1987. https://doi.org/10.3390/app14051987

[4] S. E. Asmi, T. Abar, and Asma BEN LETAIFA. 2020. Quality of experience prediction model for video streaming in SDN networks. *Int. J. Wirel. Mob. Comput.* 18 (2020), 59–70. https://doi.org/10.1504/ijwmc.2020.10026459

[5] Yang Bai, Lixing Chen, Jianhua Li, Jun Wu, Pan Zhou, Zichuan Xu, and Jie Xu. 2023. Multicore Federated Learning for Mobile-Edge Computing Platforms. *IEEE Internet of Things Journal* 10, 7 (April 2023), 5940–5952. https://doi.org/10.1109/jiot.2022.3224239

[6] Chandrasekar Balachandran, Gowri Ramachandran, Bhaskar Krishnamachari, et al. 2020. EDISON: a blockchain-based secure and auditable orchestration framework for multi-domain software defined networks. In *2020 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 144–153.

[7] A. Barakabitze, Nabajeet Barman, Arslan Ahmad, Saman Zadtootaghaj, Lingfen Sun, M. Martini, and L. Atzori. 2019. QoE Management of Multimedia Streaming Services in Future Networks: A Tutorial and Survey. *IEEE Communications Surveys & Tutorials* 22 (2019), 526–565. https://doi.org/10.1109/COMST.2019.2958784

[8] Rahime Belen-Saglam, Enes Altuncu, Yang Lu, and Shujun Li. 2023. A systematic literature review of the tension between the GDPR and public blockchain systems. *Blockchain: Research and Applications* 4, 2 (June 2023), 100129. https://doi.org/10.1016/j.bcra.2023.100129

[9] Aayush Bhansali. 2023. Cloud Security and Privacy. *International Journal for Research in Applied Science and Engineering Technology* (2023). https://doi.org/10.22214/ijraset.2023.55416

[10] Gülnaziye Bingöl, Simone Porcu, Alessandro Floris, and Luigi Atzori. 2024. WebRTC-QoE: A dataset of QoE assessment of subjective scores, network impairments, and facial &amp; speech features. *Computer Networks* 244 (May 2024), 110356. https://doi.org/10.1016/j.comnet.2024.110356

[11] Gülnaziye Bingöl, Simone Porcu, Alessandro Floris, and Luigi Atzori. 2022. QoE Estimation of WebRTC-based Audiovisual Conversations from Facial Expressions. In *2022 16th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*. 577–584. https://doi.org/10.1109/SITIS57111.2022.00092

[12] Borja Bordel Sánchez, Ramón Alcarria, Latif Ladid, and Aurel Machalek. 2024. Using Privacy-Preserving Algorithms and Blockchain Tokens to Monetize Industrial Data in Digital Marketplaces. *Computers* 13, 4 (April 2024), 104. https://doi.org/10.3390/computers13040104

[13] Khadija Bouraqia, Essaid Sabir, M. Sadik, and L. Ladid. 2019. Quality of Experience for Streaming Services: Measurements, Challenges and Insights. *IEEE Access* 8 (2019), 13341–13361. https://doi.org/10.1109/ACCESS.2020.2965099

[14] Jan Camenisch and Markus Stadler. 1997. Efficient group signature schemes for large groups. In *Annual international cryptology conference*. Springer, 410–424.

[15] Miguel Castro, Barbara Liskov, et al. 1999. Practical byzantine fault tolerance. In *OsDI*, Vol. 99. 173–186.

[16] Melissa Chase and Anna Lysyanskaya. 2006. On signatures of knowledge. In *Advances in Cryptology-CRYPTO 2006: 26th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2006. Proceedings 26*. Springer, 78–96.

[17] Weiling Chen, Fengquan Lan, Hongan Wei, Tiesong Zhao, Wei Liu, and Yiwen Xu. 2024. A comprehensive review of quality of experience for emerging video services. *Signal Processing: Image Communication* 128 (Oct. 2024), 117176. https://doi.org/10.1016/j.image.2024.117176

[18] H. Chi and A. Radwan. 2023. Fully-Decentralized Fairness-Aware Federated MEC Small-Cell Peer-Offloading for Enterprise Management Networks. *IEEE Transactions on Industrial Informatics* 19 (2023), 644–652. https://doi.org/10.1109/TII.2022.3193900

[19] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. 2016. *Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds*. Springer Berlin Heidelberg, 3–33. https://doi.org/10.1007/978-3-662-53887-6_1

[20] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. 2019. TFHE: Fast Fully Homomorphic Encryption Over the Torus. *Journal of Cryptology* 33, 1 (April 2019), 34–91. https://doi.org/10.1007/s00145-019-09319-x

[21] José Luis Corcuera Bárcena, Pietro Ducange, Francesco Marcelloni, Giovanni Nardini, Alessandro Noferi, Alessandro Renda, Fabrizio Ruffini, Alessio Schiavo, Giovanni Stea, and Antonio Virdis. 2023. Enabling federated learning of explainable AI models within beyond-5G/6G networks. *Computer Communications* 210 (Oct. 2023), 356–375. https://doi.org/10.1016/j.comcom.2023.07.039

[22] David Cuellar, Muntadher Sallal, and Christopher Williams. 2024. BSM-6G: Blockchain-Based Dynamic Spectrum Management for 6G Networks: Addressing Interoperability and Scalability. *IEEE Access* 12 (2024), 59643–59664. https://doi.org/10.1109/access.2024.3393288

[23] Morten Dahl, Daniel Demmler, Sarah El Kazdadi, Arthur Meyre, Jean-Baptiste Orfila, Dragos Rotaru, Nigel P. Smart, Samuel Tap, and Michael Walter. 2023. Noah's Ark: Efficient Threshold-FHE Using Noise Flooding. In *Proceedings of the 11th Workshop on Encrypted Computing & Applied Homomorphic Cryptography* (Copenhagen, Denmark) *(WAHC '23)*. Association for Computing Machinery, New York, NY, USA, 35–46. https://doi.org/10.1145/3605759.3625259

[24] Shuping Dang, Osama Amin, Basem Shihada, and Mohamed-Slim Alouini. 2020. What should 6G be? *Nature Electronics* 3, 1 (Jan. 2020), 20–29. https://doi.org/10.1038/s41928-019-0355-6

[25] Lívia Maria Bettini de Miranda, Rodrigo Dutra Garcia, Gowri Sankar Ramachandran, Jo Ueyama, and Fábio Müller Guerrini. 2024. Blockchain in inter-organizational collaboration: A privacy-preserving voting system for collective decision-making. *Journal of*

Information Security and Applications 85 (Sept. 2024), 103837. https://doi.org/10.1016/j.jisa.2024.103837

[26] Nathan Dillbary, Roi Yozevitch, Amit Dvir, Ran Dubin, and Chen Hajaj. 2024. Hidden in Time, Revealed in Frequency: Spectral Features and Multiresolution Analysis for Encrypted Internet Traffic Classification. In *2024 IEEE 21st Consumer Communications &amp; Networking Conference (CCNC)*. IEEE, 266–271. https://doi.org/10.1109/ccnc51664.2024.10454807

[27] Stamatia F. Drampalou, Dimitris Uzunidis, Anastasios Vetsos, Nikolaos I. Miridakis, and Panagiotis Karkazis. 2024. A User-Centric Perspective of 6G Networks: A Survey. *IEEE Access* 12 (2024), 190255–190294. https://doi.org/10.1109/ACCESS.2024.3516194

[28] Jun Du, Chunxiao Jiang, Jian Wang, Yong Ren, and M. Debbah. 2020. Machine Learning for 6G Wireless Networks: Carrying Forward Enhanced Bandwidth, Massive Access, and Ultrareliable/Low-Latency Service. *IEEE Vehicular Technology Magazine* 15 (2020), 122–134. https://doi.org/10.1109/MVT.2020.3019650

[29] Léo Ducas and Daniele Micciancio. 2015. *FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second*. Springer Berlin Heidelberg, 617–640. https://doi.org/10.1007/978-3-662-46800-5_24

[30] João Paulo Esper, Nadjib Achir, Kleber Vieira Cardoso, and Jussara M. Almeida. 2023. Impact of User Privacy and Mobility on Edge Offloading. In *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 1–6. https://doi.org/10.1109/pimrc56721.2023.10293816

[31] Tianyi Feng, Zhixiang Zhang, Wai-Choong Wong, Sumei Sun, and Biplab Sikdar. 2024. A Framework for Tradeoff Between Location Privacy Preservation and Quality of Experience in Location Based Services. *IEEE Open Journal of Vehicular Technology* 5 (2024), 428–439. https://doi.org/10.1109/ojvt.2024.3364184

[32] Mohamed Amine Ferrag, Othmane Friha, Burak Kantarci, Norbert Tihanyi, Lucas Cordeiro, Merouane Debbah, Djallel Hamouda, Muna Al-Hawawreh, and Kim-Kwang Raymond Choo. 2023. Edge Learning for 6G-Enabled Internet of Things: A Comprehensive Survey of Vulnerabilities, Datasets, and Defenses. *IEEE Communications Surveys & Tutorials* 25, 4 (2023), 2654–2713. https://doi.org/10.1109/comst.2023.3317242

[33] Mostafa M. Fouda, Zubair Md Fadlullah, Mohamed I. Ibrahem, and Nei Kato. 2024. Privacy-Preserving Data-Driven Learning Models for Emerging Communication Networks: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials* (2024), 1–1. https://doi.org/10.1109/comst.2024.3486690

[34] Yun Gao, Xin Wei, and Liang Zhou. 2020. Personalized QoE Improvement for Networking Video Service. *IEEE Journal on Selected Areas in Communications* 38, 10 (Oct. 2020), 2311–2323. https://doi.org/10.1109/jsac.2020.3000395

[35] Rodrigo Dutra Garcia. 2025. Privacy QoE System. https://github.com/rodrigodg1/privacy-qoe-system. Accessed: 2025-04-17.

[36] Craig Gentry. 2009. *A fully homomorphic encryption scheme*. Stanford university.

[37] Yanwei Gong, Xiaolin Chang, Jelena Mišić, Vojislav B. Mišić, Jianhua Wang, and Haoran Zhu. 2024. Practical solutions in fully homomorphic encryption: a survey analyzing existing acceleration methods. *Cybersecurity* 7, 1 (March 2024). https://doi.org/10.1186/s42400-023-00187-4

[38] Suyash Gupta, Jelle Hellings, and Mohammad Sadoghi. 2021. *Fault-Tolerant Distributed Transactions on Blockchain*. Springer International Publishing. https://doi.org/10.1007/978-3-031-01877-0

[39] Craig Gutterman, Katherine Guo, Sarthak Arora, Trey Gilliland, Xiaoyang Wang, Les Wu, Ethan Katz-Bassett, and Gil Zussman. 2020. Requet: Real-Time QoE Metric Detection for Encrypted YouTube Traffic. *ACM Transactions on Multimedia Computing, Communications, and Applications* 16, 2s (April 2020), 1–28. https://doi.org/10.1145/3394498

[40] Guobiao He, Wei Su, Shuai Gao, Ningchun Liu, and Sajal K. Das. 2022. NetChain: A Blockchain-Enabled Privacy-Preserving Multi-Domain Network Slice Orchestration Architecture. *IEEE Transactions on Network and Service Management* 19, 1 (March 2022), 188–202. https://doi.org/10.1109/tnsm.2021.3110057

[41] Songlin He, Qiang Tang, C. Wu, and Xuewen Shen. 2020. Decentralizing IoT Management Systems Using Blockchain for Censorship Resistance. *IEEE Transactions on Industrial Informatics* 16 (2020), 715–727. https://doi.org/10.1109/TII.2019.2939797

[42] Tharaka Hewa, Mika Ylianttila, and Madhusanka Liyanage. 2021. Survey on blockchain based smart contracts: Applications, opportunities and challenges. *Journal of Network and Computer Applications* 177 (2021), 102857. https://doi.org/10.1016/j.jnca.2020.102857

[43] Neveen Mohammad Hijazi, Moayad Aloqaily, Mohsen Guizani, Bassem Ouni, and Fakhri Karray. 2024. Secure Federated Learning With Fully Homomorphic Encryption for IoT Communications. *IEEE Internet of Things Journal* 11, 3 (Feb. 2024), 4289–4300. https://doi.org/10.1109/jiot.2023.3302065

[44] Selim Ickin, Konstantinos Vandikas, Farnaz Moradi, Jalil Taghia, and Wenfeng Hu. 2020. Ensemble-based Synthetic Data Synthesis for Federated QoE Modeling. In *2020 6th IEEE Conference on Network Softwarization (NetSoft)*. IEEE. https://doi.org/10.1109/netsoft48620.2020.9165379

[45] International Telecommunication Union. 2015. *Reference Guide to Quality of Experience Assessment Methodologies*. Recommendation ITU-T G.1011. International Telecommunication Union, Geneva, Switzerland. http://handle.itu.int/11.1002/1000/12507 Series G: Transmission Systems and Media, Digital Systems and Networks. Multimedia Quality of Service and Performance – Generic and User-Related Aspects.

[46] Abu Jahid, Mohammed H. Alsharif, and Trevor J. Hall. 2023. The convergence of blockchain, IoT and 6G: Potential, opportunities, challenges and research roadmap. *Journal of Network and Computer Applications* 217 (Aug. 2023), 103677. https://doi.org/10.1016/j.

jnca.2023.103677

[47] Khlood Jastaniah, Ning Zhang, and Mustafa A. Mustafa. 2024. Efficient User-Centric Privacy-Friendly and Flexible Wearable Data Aggregation and Sharing. *IEEE Transactions on Cloud Computing* (2024), 1–18. https://doi.org/10.1109/tcc.2024.3375801

[48] Yili Jin, Wenyi Zhang, Zihan Xu, Fangxin Wang, and Xue Liu. 2024. Privacy-Preserving Gaze-Assisted Immersive Video Streaming. *IEEE Transactions on Mobile Computing* 23, 12 (Dec. 2024), 15098–15113. https://doi.org/10.1109/tmc.2024.3452510

[49] Shahrukh Khan Kasi, U. Hashmi, S. Ekin, A. Abu-Dayya, and A. Imran. 2023. D-RAN: A DRL-Based Demand-Driven Elastic User-Centric RAN Optimization for 6G & Beyond. *IEEE Transactions on Cognitive Communications and Networking* 9 (2023), 130–145. https://doi.org/10.1109/TCCN.2022.3217785

[50] Sameera K.M., Serena Nicolazzo, Marco Arazzi, Antonino Nocera, Rafidha Rehiman K.A., Vinod P., and Mauro Conti. 2024. Privacy-preserving in Blockchain-based Federated Learning systems. *Computer Communications* 222 (June 2024), 38–67. https://doi.org/10.1016/j.comcom.2024.04.024

[51] Georgios Kougioumtzidis, Vladimir Poulkov, Zaharias D. Zaharis, and Pavlos I. Lazaridis. 2022. A Survey on Multimedia Services QoE Assessment and Machine Learning-Based Prediction. *IEEE Access* 10 (2022), 19507–19538. https://doi.org/10.1109/ACCESS.2022.3149592

[52] Aditya Kumar, Adi Bhardwaj, and Abhishek Singh. 2023. A Review of Data Security in Cloud Computing. *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (2023), 1–4. https://doi.org/10.1109/ICCCNT56998.2023.10306788

[53] Hemant Ramdas Kumbhar and S. Srinivasa Rao. 2025. Federated learning enabled multi-key homomorphic encryption. *Expert Systems with Applications* 268 (April 2025), 126197. https://doi.org/10.1016/j.eswa.2024.126197

[54] Asif Ali Laghari, Xiaobo Zhang, Zaffar Ahmed Shaikh, Asiya Khan, Vania V. Estrela, and Saadat Izadi. 2023. A review on quality of experience (QoE) in cloud computing. *Journal of Reliable Intelligent Environments* 10, 2 (June 2023), 107–121. https://doi.org/10.1007/s40860-023-00210-y

[55] Fatima Laiche, Asma Ben Letaifa, and Taoufik Aguili. 2020. QoE Influence Factors (IFs) classification Survey focusing on User Behavior/Engagement metrics. In *2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. 143–146. https://doi.org/10.1109/WETICE49692.2020.00036

[56] Lamyne, Selim Ickin, and Markus Fiedler. 2021. Poqemon-QoE-Dataset. https://github.com/Lamyne/Poqemon-QoE-Dataset. Accessed: 2024-08-22.

[57] Kim-Hung Le, Khanh-Hoi Le-Minh, and Huy-Tan Thai. 2023. BrainyEdge: An AI-enabled framework for IoT edge computing. *ICT Express* 9, 2 (April 2023), 211–221. https://doi.org/10.1016/j.icte.2021.12.007

[58] Xuanheng Li, Jiahong Liu, Xianhao Chen, Jie Wang, and Miao Pan. 2024. Caching on the Sky: A Multiagent Federated Reinforcement Learning Approach for UAV-Assisted Edge Caching. *IEEE Internet of Things Journal* 11, 17 (Sept. 2024), 28213–28226. https://doi.org/10.1109/jiot.2024.3401219

[59] Xiuhua Li, Chuan Sun, Junhao Wen, Xiaofei Wang, Mohsen Guizani, and Victor C.M. Leung. 2022. Multi-User QoE Enhancement: Federated Multi-Agent Reinforcement Learning for Cooperative Edge Intelligence. *IEEE Network* 36, 5 (Sept. 2022), 144–151. https://doi.org/10.1109/mnet.001.2200194

[60] Zhidu Li, Xuelian Gao, Qiqi Li, Jiaqi Guo, and Boran Yang. 2022. Edge Caching Enhancement for Industrial Internet: A Recommendation-Aided Approach. *IEEE Internet of Things Journal* 9 (2022), 16941–16952. https://doi.org/10.1109/JIOT.2022.3143506

[61] Ji Liu, Chunlu Chen, Yu Li, Lin Sun, Yulun Song, Jingbo Zhou, Bo Jing, and Dejing Dou. 2024. Enhancing trust and privacy in distributed networks: a comprehensive survey on blockchain-based federated learning. *Knowledge and Information Systems* (April 2024). https://doi.org/10.1007/s10115-024-02117-3

[62] Xiaozhen Lu, Liang Xiao, Pengmin Li, Xiangyang Ji, Chenren Xu, Shui Yu, and W. Zhuang. 2023. Reinforcement Learning-Based Physical Cross-Layer Security and Privacy in 6G. *IEEE Communications Surveys & Tutorials* 25 (2023), 425–466. https://doi.org/10.1109/COMST.2022.3224279

[63] Gerald Tietaa Maale, Guolin Sun, Noble Arden Elorm Kuadey, Thomas Kwantwi, Ruijie Ou, and Guisong Liu. 2023. DeepFESL: Deep Federated Echo State Learning-Based Proactive Content Caching in UAV-Assisted Networks. *IEEE Transactions on Vehicular Technology* 72, 9 (Sept. 2023), 12208–12220. https://doi.org/10.1109/tvt.2023.3268541

[64] Bomin Mao, Jiajia Liu, Yingying Wu, and Nei Kato. 2023. Security and Privacy on 6G Network Edge: A Survey. *IEEE Communications Surveys & Tutorials* 25, 2 (2023), 1095–1127. https://doi.org/10.1109/comst.2023.3244674

[65] Bomin Mao, Jiajia Liu, Yingying Wu, and Nei Kato. 2023. Security and Privacy on 6G Network Edge: A Survey. *IEEE Communications Surveys & Tutorials* 25, 2 (2023), 1095–1127. https://doi.org/10.1109/comst.2023.3244674

[66] Bomin Mao, Jiajia Liu, Yingying Wu, and Nei Kato. 2023. Security and Privacy on 6G Network Edge: A Survey. *IEEE Communications Surveys & Tutorials* 25, 2 (2023), 1095–1127. https://doi.org/10.1109/comst.2023.3244674

[67] Chiara Marcolla, V. Sucasas, Marcos Manzano, R. Bassoli, F. Fitzek, and N. Aaraj. 2022. Survey on Fully Homomorphic Encryption, Theory, and Applications. *Proc. IEEE* 110 (2022), 1572–1609. https://doi.org/10.1109/JPROC.2022.3205665

[68] Zubair Md. Fadlullah and Nei Kato. 2022. HCP: Heterogeneous Computing Platform for Federated Learning Based Collaborative Content Caching Towards 6G Networks. *IEEE Transactions on Emerging Topics in Computing* 10, 1 (Jan. 2022), 112–123. https:

//doi.org/10.1109/tetc.2020.2986238

[69] Sidra Tul Muntaha, Qasim Z. Ahmed, Faheem A. Khan, Zaharias D. Zaharis, and Pavlos I. Lazaridis. 2025. Hybrid Blockchain-Based Multi-Operator Resource Sharing and SLA Management. *IEEE Open Journal of the Communications Society* 6 (2025), 362–377. https://doi.org/10.1109/ojcoms.2024.3523362

[70] Minh-Duc Nguyen, Van Tong, Sami Souihi, and Abdelhamid Mellouk. 2023. Fully-Decentralized Federated Learning for QoE Estimation. *GLOBECOM 2023 - 2023 IEEE Global Communications Conference* 00 (2023), 5859–5864. https://doi.org/10.1109/globecom54140.2023.10437914

[71] Qinyin Ni, Zhang Linfeng, Xiaorong Zhu, and Inayat Ali. 2022. A Novel Design Method of High Throughput Blockchain for 6G Networks: Performance Analysis and Optimization Model. *IEEE Internet of Things Journal* 9, 24 (Dec. 2022), 25643–25659. https://doi.org/10.1109/jiot.2022.3194889

[72] Simone Porcu, Alessandro Floris, and Luigi Atzori. 2022. CB-FL: Cluster-Based Federated Learning applied to Quality of Experience modelling. In *2022 16th International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*. IEEE. https://doi.org/10.1109/sitis57111.2022.00093

[73] Lingjun Pu, Jianxin Shi, Xinjing Yuan, Xu Chen, Lei Jiao, Tian Zhang, and Jingdong Xu. 2024. EMS: Erasure-Coded Multi-Source Streaming for UHD Videos Within Cloud Native 5G Networks. *IEEE Transactions on Mobile Computing* 23, 2 (Feb. 2024), 1472–1487. https://doi.org/10.1109/tmc.2023.3238356

[74] Guntur Dharma Putra, Volkan Dedeoglu, Salil S. Kanhere, and Raja Jurdak. 2023. Privacy-preserving Trust Management for Blockchain-based Resource Sharing in 6G-IoT. In *2023 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. 1–9. https://doi.org/10.1109/ICBC56567.2023.10174987

[75] Bosen Rao, Jiale Zhang, Di Wu, Chengcheng Zhu, Xiaobing Sun, and Bing Chen. 2024. Privacy Inference Attack and Defense in Centralized and Federated Learning: A Comprehensive Survey. *IEEE Transactions on Artificial Intelligence* (2024), 1–22. https://doi.org/10.1109/TAI.2024.3363670

[76] Abdul Razaque, Meenhoon Khan, Joon Yoo, Aziz Alotaibi, Majid Alshammari, and Muder Almiani. 2024. Blockchain-enabled heterogeneous 6G supported secure vehicular management system over cloud edge computing. *Internet of Things* 25 (April 2024), 101115. https://doi.org/10.1016/j.iot.2024.101115

[77] Ronald L Rivest, Len Adleman, Michael L Dertouzos, et al. 1978. On data banks and privacy homomorphisms. *Foundations of secure computation* 4, 11 (1978), 169–180.

[78] Henrique Souza Rossi, Karan Mitra, Samuel Larsson, Christer Åhlund, and Irina Cotanis. 2024. Subjective QoE Assessment for Virtual Reality Cloud-based First-Person Shooter Game. In *ICC 2024 - IEEE International Conference on Communications*. 4698–4703. https://doi.org/10.1109/ICC51166.2024.10622467

[79] Velliangiri S, Rajesh Manoharan, Sitharthan Ramachandran, and Vani Rajasekar. 2022. Blockchain Based Privacy Preserving Framework for Emerging 6G Wireless Communications. *IEEE Transactions on Industrial Informatics* 18, 7 (2022), 4868–4874. https://doi.org/10.1109/TII.2021.3107556

[80] Walid Saad, Mehdi Bennis, and Mingzhe Chen. 2020. A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems. *IEEE Network* 34, 3 (2020), 134–142. https://doi.org/10.1109/MNET.001.1900287

[81] Chamara Sandeepa, Bartlomiej Siniarski, Nicolas Kourtellis, Shen Wang, and Madhusanka Liyanage. 2024. A Survey on Privacy of Personal and Non-Personal Data in B5G/6G Networks. *Comput. Surveys* 56, 10 (June 2024), 1–37. https://doi.org/10.1145/3662179

[82] Paul Scalise, Matthew Boeding, Michael Hempel, Hamid Sharif, Joseph Delloiacovo, and John Reed. 2024. A Systematic Survey on 5G and 6G Security Considerations, Challenges, Trends, and Research Areas. *Future Internet* 16, 3 (Feb. 2024), 67. https://doi.org/10.3390/fi16030067

[83] Marialisa Scatá and Aurelio La Corte. 2023. A Complex Insight for Quality of Service Based on Spreading Dynamics and Multilayer Networks in a 6G Scenario. *Mathematics* (2023). https://doi.org/10.3390/math11020423

[84] R. Sendhil and A. Amuthan. 2022. Verifiable quaternion fully homomorphic encryption scheme for mitigating false data injection attacks by privacy preservation in fog environment. *Journal of Information Security and Applications* 71 (Dec. 2022), 103383. https://doi.org/10.1016/j.jisa.2022.103383

[85] Mehdi Setayesh and Vincent W.S. Wong. 2024. Viewport Prediction, Bitrate Selection, and Beamforming Design for THz-Enabled 360-Degree Video Streaming. *IEEE Transactions on Wireless Communications* (2024), 1–1. https://doi.org/10.1109/twc.2024.3513221

[86] Ahmed Shafee, S.R. Hasan, and Tasneem A. Awaad. 2025. Privacy and security vulnerabilities in edge intelligence: An analysis and countermeasures. *Computers and Electrical Engineering* 123 (April 2025), 110146. https://doi.org/10.1016/j.compeleceng.2025.110146

[87] Nafiseh Sharghivand, F. Derakhshan, Lena Mashayekhy, and Leyli Mohammadkhanli. 2022. An Edge Computing Matching Framework With Guaranteed Quality of Service. *IEEE Transactions on Cloud Computing* 10 (2022), 1557–1570. https://doi.org/10.1109/TCC.2020.3005539

[88] Ankita Tondwalkar, Pilar Andres-Maldonado, Devaki Chandramouli, Rainer Liebhart, Fernando Sanchez Moya, Troels Kolding, and Pablo Perez. 2024. Provisioning Quality of Experience in 6G Networks. *IEEE Access* 12 (2024), 127007–127017. https://doi.org/10.1109/ACCESS.2024.3455938

[89] H. Tran, S. Hoceini, A. Mellouk, Julien Perez, and S. Zeadally. 2014. QoE-Based Server Selection for Content Distribution Networks. *IEEE Trans. Comput.* 63 (2014), 2803–2815. https://doi.org/10.1109/TC.2013.33

[90] Dmitrii Usynin, Daniel Rueckert, and Georgios Kaissis. 2023. Beyond Gradients: Exploiting Adversarial Priors in Model Inversion Attacks. *ACM Transactions on Privacy and Security* 26, 3 (June 2023), 1–30. https://doi.org/10.1145/3592800

[91] Xin Wang, Achyut Shankar, Keqin Li, B. D. Parameshachari, and Jianhui Lv. 2024. Blockchain-Enabled Decentralized Edge Intelligence for Trustworthy 6G Consumer Electronics. *IEEE Transactions on Consumer Electronics* 70, 1 (2024), 1214–1225. https://doi.org/10.1109/TCE.2024.3371501

[92] Yichuan Wang, Xiaolong Liang, Xinhong Hei, Wenjiang Ji, and Lei Zhu. 2021. Deep Learning Data Privacy Protection Based on Homomorphic Encryption in AIoT. *Mobile Information Systems* 2021 (June 2021), 1–11. https://doi.org/10.1155/2021/5510857

[93] Feng Xia and Wenhao Cheng. 2024. A survey on privacy-preserving federated learning against poisoning attacks. *Cluster Computing* 27, 10 (July 2024), 13565–13582. https://doi.org/10.1007/s10586-024-04629-7

[94] Qichao Xu, Zhou Su, and Q. Yang. 2020. Blockchain-Based Trustworthy Edge Caching Scheme for Mobile Cyber-Physical System. *IEEE Internet of Things Journal* 7 (2020), 1098–1110. https://doi.org/10.1109/JIOT.2019.2951007

[95] Jianbin Xue, Jia Yao, and Jiahao Wang. 2024. A dynamic pricing scheme for secure offloading and resource allocation based on the internet of vehicles. *Ad Hoc Networks* 161 (Aug. 2024), 103545. https://doi.org/10.1016/j.adhoc.2024.103545

[96] Yinghao Yang, Huaizhi Zhang, Shengyu Fan, Hang Lu, Mingzhe Zhang, and Xiaowei Li. 2023. Poseidon: Practical Homomorphic Encryption Accelerator. In *2023 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. 870–881. https://doi.org/10.1109/HPCA56546.2023.10070984

[97] Jannah Yusoff, Zarina Mohamad, and Mohd Anuar. 2022. A Review: Consensus Algorithms on Blockchain. *Journal of Computer and Communications* (2022). https://doi.org/10.4236/jcc.2022.109003

[98] Zama. 2023. Confidential EVM Smart Contracts using Fully Homomorphic Encryption. https://github.com/zama-ai/fhevm/. Accessed: 2023-12-13.

[99] Zama. 2024. FHEVMJS. https://github.com/zama-ai/fhevmjs Accessed: 2024-08-28.

[100] Zama. 2024. tfhe-rs: A Rust library for fully homomorphic encryption. https://github.com/zama-ai/tfhe-rs. Accessed: 2024-08-26.

[101] Zizhen Zhang, Tengfei Cao, Xiaoying Wang, Han Xiao, and Jianfeng Guan. 2022. VC-PPQ: Privacy-Preserving Q-Learning Based Video Caching Optimization in Mobile Edge Networks. *IEEE Transactions on Network Science and Engineering* 9, 6 (Nov. 2022), 4129–4144. https://doi.org/10.1109/tnse.2022.3195926

[102] V. Ziegler, Peter Schneider, H. Viswanathan, M. Montag, S. Kanugovi, and Ali Rezaki. 2021. Security and Trust in the 6G Era. *IEEE Access* 9 (2021), 142314–142327. https://doi.org/10.1109/access.2021.3120143

[103] Yiping Zuo, Jiajia Guo, Ning Gao, Yongxu Zhu, Shi Jin, and Xiao Li. 2023. A Survey of Blockchain and Artificial Intelligence for 6G Wireless Communications. *IEEE Communications Surveys & Tutorials* 25, 4 (2023), 2494–2528. https://doi.org/10.1109/comst.2023.3315374