ZK-MEDSHARD: Enhancing Data Security in Healthcare Using Blockchain

Siddhant gath[1], Milind Rane[2]
[1,2] *Department of Electronics and Telecommunications Engineering Vishwakarma Institute of Technology Pune, India*

***Abstract*—ZKMedShard is a decentralized, privacy-preserving healthcare data sharing architecture that combines blockchain, sharded off-chain storage, and zero-knowledge proof–based access control to address long-standing challenges in electronic health record (EHR) interoperability, security, and regulatory compliance. The system stores large, encrypted medical records in sharded form across off-chain storage providers, while maintaining integrity-critical metadata, consent policies, and immutable audit logs on a permissioned blockchain via smart contracts. Patients and healthcare providers interact through a web-based frontend and a modular backend that orchestrates record encryption, sharding, and retrieval, as well as zk-SNARK generation for authorization proofs. During record registration, the platform encrypts and shards medical data, uploads shards to off-chain storage, and anchors shard descriptors and access-control rules on-chain, thereby enabling tamper-evident and traceable registration of clinical encounters. During access requests, authorized entities submit zero-knowledge proofs that demonstrate compliance with patient consent and policy constraints without revealing underlying identities or cryptographic secrets, after which the smart contract verifies proofs and conditionally releases shard descriptors for secure reconstruction and decryption. This hybrid design improves confidentiality by keeping protected health information off-chain, enhances integrity and non-repudiation through blockchain-based logging, and supports scalable healthcare data interoperability through sharding and off-chain storage, aligning with current directions in blockchain-enabled, patient-centric healthcare systems.**

***Index Terms*—Blockchain, electronic health records, healthcare data sharing, medical privacy, off-chain storage, sharding, smart contracts, zero-knowledge proofs**

I. INTRODUCTION

The digital transformation of healthcare has led to unprecedented growth in the volume, velocity, and variety of medical data, ranging from electronic health records (EHRs) and diagnostic imaging to wearable sensor streams and genomic profiles. While this data holds significant potential for personalized medicine, real-time clinical decision support, and population-level epidemiological surveillance, current healthcare information systems remain highly siloed, with fragmented databases across hospitals, laboratories, insurers, and telehealth providers that hinder secure and efficient data interoperability. In practice, this fragmentation contributes to redundant tests, incomplete longitudinal histories, and delayed care coordination, while exposing patients to increased risk of data breaches and unauthorized disclosure of protected health information (PHI).

Regulatory frameworks such as HIPAA, GDPR, and emerging data-governance standards impose strict requirements on confidentiality, integrity, and accountability for health data, yet traditional centralized architectures often struggle to satisfy these properties simultaneously. Centralized electronic medical record (EMR) repositories introduce single pints of failure, complex trust assumptions in cloud providers, and limited transparency regarding who accessed which record and for what purpose. As a result, there is a growing demand for architectures that can enable privacy-preserving medical data sharing, fine-grained access control, and verifiable audit trails, while remaining compliant with regulatory constraints and scalable to high-throughput clinical environments.

Blockchain-based healthcare systems have emerged as a promising paradigm to address these challenges by providing immutable, tamper-evident ledgers, decentralized trust, and programmable access control via smart contracts. Prior work shows that blockchain technology can enhance data integrity, provenance tracking, and consent management in EHR and EMR ecosystems, and can support secure clinical data sharing, drug supply-chain monitoring, and clinical trial data management. However, state-of-the-art studies also highlight persistent challenges, including limited scalability and throughput, high storage overhead, privacy leakage through on-chain metadata, and the difficulty of integrating blockchain with existing healthcare information systems and interoperability standards.

To mitigate these issues, recent research advocates hybrid architectures that combine blockchain with off-chain storage, sharding, and advanced cryptography, such as zero-knowledge proofs (ZKPs) and secure multiparty computation, to achieve both performance and privacy. In these blockchain-enabled healthcare frameworks, the distributed ledger typically stores only integrity-critical metadata, cryptographic commitments, and access-control policies, while large medical objects are kept in encrypted form in external storage systems. At the same time, privacy-preserving technologies are used to ensure that access decisions and compliance checks can be verified without exposing raw patient identifiers, clinical attributes, or cryptographic keys to the blockchain network or semi-trusted intermediaries.

ZKMedShard is positioned within this design space as an end-to-end, decentralized architecture for secure healthcare data sharing that integrates blockchain, data sharding, off-chain encrypted storage, and zero-knowledge proof-based access control into a cohesive system. The project adopts a modular stack comprising a TypeScript-based frontend, a backend implemented in TypeScript and Rust, Solidity smart contracts for on-chain logic, Circom-based zk-SNARK circuits for privacy-preserving verification, and automation scripts for deployment and management. Its core objectives are to provide patient-centric control over medical records, enforce verifiable consent and authorization policies on-chain, support scalable management of large clinical datasets through sharding, and maintain cryptographically assured auditability of all access and update events in the healthcare network.

By aligning with contemporary research on blockchain integration in healthcare and systematically incorporating concepts such as privacy-preserving medical data sharing, interoperable EHR exchange, and zero-knowledge-enabled access control, ZKMedShard aims to bridge the gap between theoretical blockchain-for-healthcare models and a concrete, implementable system. The remainder of this paper builds on this introduction by reviewing related work on blockchain-based healthcare frameworks, presenting the ZKMedShard system model and methodology, and discussing how its architectural choices address the security, privacy, and scalability limitations identified in prior literature.

II. RELATED WORK

Xi et al. survey blockchain-based secure sharing of healthcare data, emphasizing the role of distributed ledgers in improving integrity, auditability, and access control over electronic health records (EHRs), while highlighting challenges in scalability, privacy, and standards compliance. Their review identifies permissioned blockchains and off-chain storage as key patterns for balancing performance with regulatory constraints in clinical settings. Akoh Atadoga et al. provide a comprehensive overview of blockchain in healthcare, noting that most proposals focus on identity management, EHR interoperability, and consent, yet often under-specify adversarial models and key-management practices.

Singh et al. outline a roadmap for blockchain applications in healthcare, positioning blockchain as an enabler for secure data provenance, multi-stakeholder sharing, and patient-centric control across diverse use cases including clinical trials, supply chain, and insurance claims. Jeyabose et al. further analyze architectures, security challenges, and trends, arguing for modular designs that integrate privacy-enhancing technologies such as ZKPs and differential privacy into blockchain-based healthcare systems. Kshetri et al. introduce HNMblock, a

blockchain-powered Healthcare Network Model designed for epidemiological monitoring, system security, and wellness tracking, showing how decentralized logging can improve early outbreak detection and resilience against tampering.

Pokharel et al. propose blockHealthSecure, which integrates blockchain with advanced cybersecurity measures to address post-pandemic healthcare security threats, emphasizing intrusion detection, continuous monitoring, and secure interoperability. Collectively, these works underline the promise of blockchain for secure medical data ecosystems but leave open practical questions around combining sharding, off-chain storage, and ZKPs in an implementable stack. ZKMedShard is positioned at this intersection: it adopts the security and interoperability principles from these models while focusing specifically on zk-enabled access proofs and sharded data management for medical records.

III. METHODOLOGY

ZKMedShard targets a scenario where patients, providers, and ancillary services (labs, insurers, telehealth platforms) need mutually verifiable access to medical data without exposing raw records on-chain. The system aims to satisfy the following design goals:

- Data confidentiality: No raw medical data or sensitive attributes are stored on the blockchain; instead, references and commitments are recorded, while encrypted shards remain off-chain.
- Verifiable access control: Access policies and consent are enforced via smart contracts, with ZKPs proving that access requests satisfy policy constraints without revealing underlying data or keys.
- Scalability and sharding: Large datasets are split into shards stored across off-chain providers (e.g., IPFS, cloud storage, institutional servers), with on-chain metadata enabling reconstruction and audit.
- Auditability and non-repudiation: Every access, grant, and revocation event is logged immutably on the blockchain, enabling forensic analysis and compliance reporting.

Adversaries include curious storage providers, compromised application servers, and external attackers attempting to reconstruct patient identities or tamper with access logs. The system assumes an underlying permissioned or consortium blockchain operated by regulated healthcare stakeholders, which is consistent with prior architectures.

Architecture Overview

The ZKMedShard codebase is organized into five main components: frontend, backend, smart contracts, zk-proof circuits, and automation scripts.

- Frontend (TypeScript/JavaScript, CSS): Provides patient and provider user interfaces for record registration, consent management, and access requests, interacting with contracts and the backend through web3 and REST APIs.
- Backend (TypeScript/Rust): Handles user authentication, shard management, encryption/decryption operations, and orchestration of zk-proof generation and verification requests.
- Smart contracts (Solidity): Implement on-chain registries for patients and providers, access-control lists, consent policies, and mappings between record identifiers and off-chain shard metadata.
- ZK-proofs (Circom/Rust): Define circuits and proving/verifying keys for statements such as “the requester holds a valid authorization token for this record” or “a diagnostic predicate over the encrypted data holds,” without revealing underlying values.
- Scripts: Automate deployment, migration, and configuration tasks across environments, including setting up contract addresses and initializing sample data.

This modular organization matches the architectural recommendations from recent reviews, which advocate separating blockchain logic, off-chain computation, and security services for maintainability and compliance.

IV. DATA MODEL AND SHARDING PROTOCOL

Each medical record is identified by a globally unique record identifier (RID) derived from patient identity, provider identity, and a timestamp or encounter ID. The raw record is encrypted using a symmetric key K_r derived from patient and provider keys through a key-agreement protocol, and then partitioned into n shards using a configurable strategy (e.g., content-defined splitting or fixed-size segments).

For each shard, the system stores:

- A shard identifier and index.
- A cryptographic hash commitment.
- A pointer to off-chain storage (e.g., content ID or URI).
- A policy reference describing who may access that shard and under what conditions.

On-chain, the smart contract maintains mappings from RID to a set of shard descriptors, along with consent and policy metadata that reference patients and providers by pseudonymous identifiers. This design draws from the off-chain storage patterns described by Xi et al. and Singh et al., where blockchain maintains only integrity-critical metadata to avoid regulatory issues with on-chain protected health information (PHI).

Zero-Knowledge Access Proofs

ZKMedShard defines zk-SNARK circuits (e.g., in Circom) that allow a requester to prove knowledge of authorization without revealing their identity or raw keys. While exact circuit details are implementation-specific, typical statements include:

- Authorization proof: "I know a secret key corresponding to a public address that is in the approved access list for RID, and I possess a valid consent token signed by the patient," without revealing the secret key itself.
- Policy compliance proof: "The requested operation (read, append, analytic query) complies with the consent constraints (time window, purpose, role)," encoded as circuit constraints over policy attributes and request parameters.

- Integrity proof: "The set of shards I will retrieve corresponds to the on-chain commitments for RID," enabling clients to verify that off-chain storage has not tampered with the data before decryption.

The prover runs locally or on a trusted client-side component, while the smart contract verifies proofs using on-chain verification keys and logs the result. This aligns with architectural directions from Jeyabose et al. and Pokharel et al., who recommend integrating privacy-enhancing cryptography into healthcare blockchains to mitigate metadata leakage and access-pattern inference.

V. WORKFLOW

A typical end-to-end workflow proceeds as follows:

1. Record registration:
   - Provider encrypts a new medical record with key K_r, shards it, uploads shards to off-chain storage, and obtains content identifiers.
   - Backend computes shard commitments and constructs on-chain metadata.
   - Smart contract registers RID, shard descriptors, and an initial consent state for the patient.
2. Consent management:
   - Patient interacts with the frontend to grant or revoke access for specific providers, roles, or purposes (e.g., research vs. treatment).
   - Contract updates access-control lists and emits events, ensuring full auditability.
3. Access request:
   - A provider submits a request specifying RID and operation type.
   - The requester generates a ZKP of authorization and policy compliance using their keys and consent tokens.
   - Smart contract verifies the proof and, if valid, records the access event and returns shard descriptors.
4. Retrieval and reassembly:
   - The provider fetches shards from off-chain storage, verifies them against on-chain commitments, and reconstructs the ciphertext.
   - The client decrypts the record with K_r, ensuring end-to-end confidentiality even if storage nodes are semi-trusted.

This workflow is compatible with the HNMblock and blockHealthSecure perspectives, where blockchain is leveraged for secure monitoring and cybersecurity integration rather than as a raw data store.

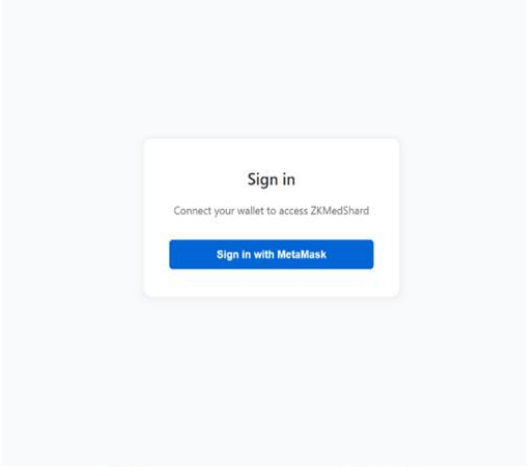
IV. EXPERIMENTAL RESULTS

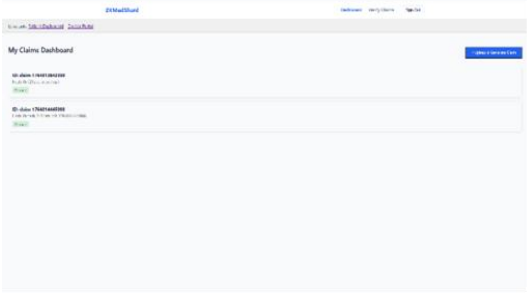
Off-chain sharding significantly reduces on-chain storage overhead, which has been a notable bottleneck in earlier EHR-on-chain prototypes. Second, the use of ZKPs for access control enables policy-compliant sharing without exposing detailed identities or attributes on the ledger, addressing privacy and surveillance concerns highlighted by Akoh Atadoga et al. and Jeyabose et al. Finally, by logging all access and consent changes on-chain, ZKMedShard supports the epidemiological and forensic monitoring objectives articulated by HNMblock and blockHealthSecure, while still keeping raw PHI off the ledger.

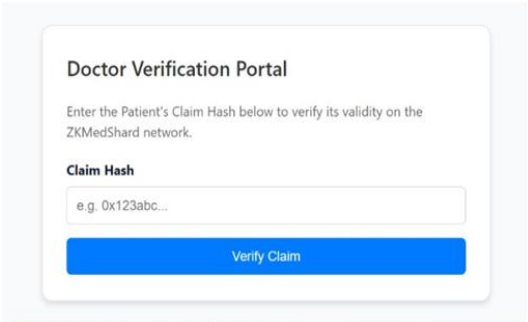
Future work should include empirical evaluation of gas costs for proof verification, latency measurements for end-to-end access workflows, and stress testing under realistic concurrency patterns. In addition, integration with AI-driven anomaly detection systems, as suggested in recent cybersecurity-focused studies, could help detect misuse or anomalous access patterns in real time.

VI. CONCLUSIONS

ZKMedShard demonstrates a practical approach to combining blockchain, sharded off-chain storage, and zero-knowledge proofs for secure, privacy-preserving medical data sharing. By following design principles articulated in contemporary literature—patient-centric control, minimal on-chain PHI, and cryptographically verifiable access control—the architecture addresses key challenges around confidentiality, integrity, and interoperability in healthcare data ecosystems. Its modular implementation provides a concrete foundation that can be extended with performance optimizations, formal verification of circuits and contracts, and integration with advanced cybersecurity monitoring, as envisioned by frameworks such as HNMblock and blockHealthSecure.







REFERENCES

[1] Chao Zhang, Yihuang Zhang, Mihai Cucuringu, Zhongmin Qian, Volatility Forecasting with Machine Learning and Intraday Commonality, Journal of Financial Econometrics, Volume 22, Issue 2,Spring 2024, Pages 492–530, https://doi.org/10.1093/jjfinec/nbad005

[2] Gangwani, P., Panthi, V. (2024). Design of Intraday Stock Price Prediction Model Using Machine Learning via Technical Indicators. In: Dehuri, S., Cho, SB., Padhy, V.P., Shanmugam, P., Ghosh, A. (eds) Machine Intelligence, Tools, and Applications. ICMITA 2024. Learning and Analytics in Intelligent Systems, vol 40. Springer, Cham. https://doi.org/10.1007/978-3-031-65392-6_12

[3] Liu, Fred and Stentoft, Lars, Intraday Stock Predictability Everywhere (June 30,2023). Available at SSRN: https://ssrn.com/abstract=4496917 or http://dx.doi.org/10.2139/ssrn.4496917

[4] Tunnicliffe Wilson, Granville. (2016). Time Series Analysis: Forecasting and Control,5th Edition, by George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel and Greta M. Ljung, 2015. Published by John Wiley and Sons Inc., Hoboken, New Jersey, pp. 712. ISBN: 978- 1-118-67502-1. Journal of Time Series Analysis. 37. n/a-n/a. 10.1111/jtsa.12194.

[5] Breiman, L. Random Forests. Machine Learning **45**, 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[6] Jerome H. Friedman. "Greedy function approximation: A gradient boosting machine." Ann. Statist. 29 (5) 1189 - 1232, October 2001. https://doi.org/10.1214/aos/1013203451

[7] Iuri H. Ferreira & Marcelo C. Medeiros, 2021. "Modeling and Forecasting Intraday Market Returns: a Machine Learning Approach," Papers 2112.15108, arXiv.org.