International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Enhancing Cloud Data Security Using Hyperledger

Anuja Chincholkar, Ankit raj, Aniket pawar, Pratik varat
School of Computing, MIT ADT University, Pune, India

Abstract - Cloud computing has changed the digital infrastructure by allowing scalability and on demand access to shared computing resources. Nevertheless, the traditional cloud designs present centralized systems that are targeted by the constant dangers of unauthorized access, data manipulation, single-point-of-failure, and inadequate auditability. Hyperledger Fabric with the help of blockchain technology is an attractive solution to these security issues due to its decentralized, permissioned, and tamper-evident structure. In this paper, the author suggests a new architecture that can be used to implement Hyperledger Fabric in cloud computing to improve data security, confidentiality, and integrity. The suggested architecture uses the access control on a fine-grained basis with the help of chaincode-based smart contracts, data confidentiality through the use of AES-256 encryption, and an audit trail that is immutable and allows a regulatory body to monitor an activity, as well as a forensic trace. Experimental analyses on an implemented system on a Docker-based simulation platform show that the suggested system can support throughput of 1,200 transactions per second (TPS) and average latency of 420 milliseconds at a moderate load. The relative analysis with the existing cloud security solutions proves that the proposed framework decreases the level of unauthorized access by 94.7 percent and the risk of data tampering by 98.2 percent. The findings confirm that Hyperledger Fabric integration is very effective in enhancing cloud data security without compromising operational efficiency, and thus, it can be adopted to perform enterprise-scale deployments.

Keywords - ***Cloud Security; Hyperledger Fabric; Blockchain; Smart Contracts; Access Control; Data Integrity;*** **Chaincode**

I. INTRODUCTION

A high rate of explosion of cloud computing has is fundamentally changing the way organizations store, process and share data in industries. Buyya et al. [16] envisaged cloud computing as the fifth utility, that provides computing resources on demand and scale, this vision became a reality in the form of a multi trillion dollar industry all around the world. The most recent market prognosis granted by MarketsandMarkets [1] suggests that the world market in cloud computing will be expanded significantly in the year 2030 due to the increased rate of enterprise uptake in the areas of healthcare, finance, education and logistics. Examples of the various processes currently being placed in the cloud infrastructure include cloud-hosted eBook management systems [8, 21] and AI-assisted cloud-based note-taking systems [12].

Regardless of its potent transformative strengths, cloud computing creates a multifaceted and ever-changing threat environment. The Cloud Security Alliance [2] lists eleven categories of serious cloud threat categories, generally known as the Pandemic 11, among which are: unauthorized access, data tampering, replay attacks, man in the middle (MITM) interception, and audit trail compromise. The conventional centralized cloud security design is conceptually ineffective in protecting against these threats at scale. They depend on one, reliable administrative body, making the system vulnerable on a structural level, which can be used by the enemies. The necessity to have a data dynamic and public auditing of a cloud storage dates back to 2011, when Wang et al. [14] detected that no single provider is sufficient to verify the integrity of data.

The blockchain technology has since been structurally a good solution to these restrictions. The white paper produced in 2008 by Nakamoto [3] has led to a decentralized, tamper-proof registry based on the Bitcoin protocol, and has laid the groundwork of distributed consensus, cryptographic chaining, and inscribing a record that cannot be altered or destroyed. One of the first to describe that these qualities have little to do with cryptocurrency was Crosby et al. [18], and blockchain can be used in supply chains, identity management, and data security. Zheng et al. [20] offer an official architectural overview of the blockchain systems, defining the consensus mechanisms, transaction models, and trends of scalability in the future. As early as 2015, Zyskind et al. [11] have shown that blockchain architecture could be explicitly used to decentralize privacy and secure personal data against disclosure or abuse by unauthorized parties.

Crossing the blockchain and cloud security points, scholars have put forward various solutions that are specific. Xu et al. [4] have created a blockchain-based cloud storage auditing scheme known as DASE, which does not require the use of a trusted third-party auditor. Ali et al. [6] conducted a survey of the application of smart contracts to implement security and privacy policies in cloud-based systems and found the feasibility of programmable and self-executing access control. Christidis and Devetsikiotis [15] applied the concept of blockchain-smart contract to IoT settings, whereas Dorri et al. [5] confirmed its usefulness by applying it to the case

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

of a smart home security setting. All of these works prove the idea that the implementation of blockchain into the cloud security architecture can help to resolve the fundamental vulnerabilities of centralized models.

The healthcare sector is a good example of such stakes. Hyperledger Fabric, in particular, was considered by Kumar and Mallick [9] when discussing the security issues in healthcare data management. Jiang et al. [10] introduced BlocHIE, a working blockchain-based healthcare information exchange, confirming the fact that these systems may be scaled without losing the data confidentiality. Salah et al. [19] expanded the scope even further when it comes to blockchain as an essential infrastructure element in having AI data pipelines, in which the notions of provenance and integrity are considered prerequisites to reliable model outputs.

In the case of enterprises, we find published constraints in throughput, latency and confidentiality in public blockchain networks like Bitcoin and Ethereum. Hyperledger Fabric, proposed by Androulaki et al. [13, 23] and architecturally defined by Cachin [22], is the solution to these limitations based on the modular, permissioned design. Its configuration of privacy through its separation of endorsement, ordering and validation stages allows it to have high throughput. Thakkar et al. [7] strictly benchmarked the performances of Hyperledger Fabric regarding performance under different configurations, and critical optimization parameters were determined. Li et al. [17] examined the tradeoffs between scalability and privacy in the deployment of blockchain to industries, and Vukolić [24] offered a theoretical support behind the idea that BFT-based consensus is preferable to Proof-of-Work in permissioned enterprise blockchains.

Based on these results, this paper suggests the use of SecureCloud-HLF, a cloud-based security system with blockchain applications on Hyperledger Fabric. A single end-to-end permissioned blockchain architecture is expected to reduce the five most critical categories of threats identified by the CSA [2] unauthorized access, data tampering, replay attacks, MITM interception, and audit trail compromise. As it is experimentally proved, the detection rates of the SecureCloud-HLF are over 94% in all the types of threats, and throughput with read-heavy workloads is up to 1,420 TPS, as well as the success rate with all the workload profiles is over 99%.

The rest of the paper is structured as follows: Section II is the review of related work. Section III defines the system architecture proposal. The implementation is shown in Section IV. Experimental examination and findings are in section V. Section VI is rounded off with future research directions.

II. LITERATURE REVIEW

Cloud computing: Foundations and security issues.

Buyya et al. [16] defined the conceptual background of cloud computing as a utility grade computing paradigm, which is a network of virtualized computers that are on-demand and selling them as a metered service. Their labor has discovered the fundamental service models Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) that have formed the base of the entire cloud industry globally. The MarketsandMarkets prediction [1] confirms that this industry is still experiencing a growing accelerating rate up till 2030 with security being a concern of growing economic and operational concern.

The scope of cloud-hosted application is growing very broad. Chincholkar et al. [8, 21] have shown a cloud-hosted eBook management system based on Firebase to provide a secure and efficient storage, and Chincholkar et al. [12] have shown a cloud-based and NLP-based summarization system with AI to take interactive notes. This set of applications underlines the fact that state-of-the-art cloud-based solutions need to support a variety of data types and accessing patterns and provide a high level of security assurance.

The Cloud Security Alliance [2] offers the most extensive list of the existing cloud vulnerabilities in the form of their report named Pandemic 11. The named threats such as data breaches, misconfiguration, insecure interfaces, account hijacking, and insider threats display not only the complexity nature of a cloud environment but also the ineffectiveness of a strictly perimeter-based protection. Wang et al. [14] considered the particular issue of storage security, and offered the scheme of public auditability and data dynamics, according to which cloud clients can ensure the integrity of outsourced data without access, in full. Their work defined the technical needs — correctness, unforgeability, and efficiency — that any cloud data integrity scheme needs to have, which continue to be the focus of the current work.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

An architecture and principles of blockchain Technology.

The blockchain paradigm can be traced back to the bitcoin whitepaper by Nakamoto [3], that proposed a decentralized registry glued together by cryptographic proof-of-work, and allowed peer-to-peer commerce without trust. Although Bitcoin has defined the basic model, Crosby et al. [18] elucidated the wider applicability of blockchain properties, immutability, decentralization and transparency, to non-financial applications such as supply chain management, identity verification, data integrity assurance.

Zheng et al. [20] have presented an in-depth technical description of blockchain architecture, categorized consensus mechanisms (Proof-of-Work, Proof-of-Stake, Delegated PoS, and BFT variants), which defines limitations on throughput of transactions, and estimates future development to achieve more scalable and privacy-preserving design. Their taxonomy is extensively referenced in order to compare blockchain platforms in terms of their reference framework. In their work, Zyskind et al. [11] have shown one of the first and most impactful uses of blockchain to protect privacy, where access control policies with personal data are coded directly on a blockchain so that there is neither a single point of control nor failure.

Vukolić [24] offered a strict theoretical investigation of the scalability constraints of Proof-of-Work consensus and claimed that the Byzantine Fault Tolerant replication will be a better throughput and finality assurance in permissioned environments. The design decisions of later enterprise blockchain systems, such as Hyperledger Fabric, were based on this analysis. Salah et al. [19] further applied the blockchain debate to the AI sphere and conducted a review of the available issues surrounding the application of blockchain to ensure the provenance, integrity, and auditability of training datasets and model parameters.

Hyperledger Fabric: Permissioned Blockchain for Enterprise

The most popular permissioned blockchain platform used in enterprises is the Hyperledger Fabric [13, 23] developed under the Hyperledger umbrella of Linux Foundation. Androulaki et al. presented its new execute-order-validate architecture, which splits the transaction lifecycle into three different stages performed by various participants of the network. This isolation allows simultaneous smart contract execution (chaincode), as well as configurable ordering services, and fine-grained endorsement policy - all of which allow much greater throughput compared to public blockchain networks. The first architectural specification of the Hyperledger Fabric design was given by Cachin [22], which defined the basis of cryptographic and distributed systems on which the platform is developed.

Thakkar et al. [7] performed a systematic performance benchmarking of Hyperledger Fabric and tested the effect of the complexity of the endorsement policy, block size, channel configuration, and CouchDB versus LevelDB state databases on the transaction throughput and latency. Their findings pointed out that the performance of the job of endorsing policy and access to state databases are the main bottlenecks in their performance, which can be used to offer concrete configuration advice to high-throughput deployments. Li et al. [17] examined tradeoffs in privacy and scalability of industrial Hyperledger deployments by suggesting a method of providing transaction confidentiality but not auditability, a tradeoff that is directly applicable to cloud security applications.

Blockchain for Cloud Security and Data Integrity

The use of blockchain in cloud data security has generated an increasing amount of special contributions. The authors Xu et al. [4] suggested a blockchain-based data auditing scheme, DASE, that enables cloud clients to check the integrity of data stored on a cloud by executing a challenge-response protocol based on a smart-contract. DASE removes the previously assumed trusted third-party auditor of the previous schemes replacing centralized trust with cryptographically verifiable blockchain records. The theoretical framework of provable data possession, and public auditability, as described by Wang et al. [14], was previously developed, and now provided by blockchain-enforced assurances by DASE and other schemes.

The intersection of smart contracts and cloud security was surveyed by Ali et al. [6] and schemes of access control enforcement, data confidentiality, and accountability logging were listed. Their survey indicated the access control based on smart contract to be the most mature area of application, and observed open challenges in key management, gas cost optimization and cross-chain interoperability. The survey defines the large design space that the current work belongs to.

Blockchain for IoT and Smart Environments

The contribution of Christidis and Devetsikiotis [15] is seminal, as they showed that blockchain and smart contracts are a natural trust layer in the IoT networks, i.e., autonomous device-to-device transactions and tamper-resistant audit logs can be achieved

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

without centralized intermediaries. Specifically, their work defined the peculiar features of the IoT resource-limited, intermittent-connected, heterogeneous device population that blockchain architectures need to support. Dorri et al. [5] confirmed these principles using a real-life case of a smart home, suggesting a lightweight blockchain framework where a local miner is used to manage the access control policies of IoT devices in the house, communicating with the world blockchain to provide audit and accountability. Their system proved that even in the resource-constrained environments, a blockchain-based security proves to be practical.

Blockchain in Healthcare Data Management

The healthcare industry is one of the most challenging cases when it comes to data protection due to the sensitivity of patient records, the characteristics of multi-institutional data exchange, and the high-regulatory standards. Kumar and Mallick [9] also discussed the particular relevance of Hyperledger Fabric to healthcare security issues, tested its operation under the conditions of healthcare workload distribution, and proved that the permissioned, identity-aware architecture of Hyperledger Fabric fits the requirements of the HIPAA-style compliance through its functionality. Jiang et al. [10] introduced a working blockchain-based healthcare information exchange platform, BlocHIE that confirmed the potential of blockchain to facilitate real-time and cross-institutional data sharing without affecting the privacy of patients or the integrity of their data. These domain-specific checks facilitate the application of blockchain-based security structures to regulate and sensitive cloud contexts in general.

Research Gaps and Motivation

The surveyed literature collectively establishes three important observations. First, conventional cloud security architectures remain vulnerable to the threat categories catalogued by the CSA [2], particularly those involving data tampering, unauthorized access, and audit trail manipulation. Second, blockchain technology — especially permissioned platforms such as Hyperledger Fabric [13, 22, 23] — provides the cryptographic and architectural primitives necessary to address these vulnerabilities through immutable logging, smart-contract-enforced access control, and decentralized trust. Third, while prior works have addressed individual threat vectors in isolation [4, 5, 6, 14, 15], no existing work provides a unified, empirically evaluated framework that addresses all five critical CSA threat categories simultaneously within a Hyperledger Fabric-based cloud security architecture.

SecureCloud-HLF is proposed to fill this gap. By integrating Hyperledger Fabric's permissioned blockchain with a cloud security middleware layer, the proposed system provides end-to-end protection against unauthorized access, data tampering, replay attacks, MITM interception, and audit trail compromise — validated through comprehensive performance evaluation across multiple workload profiles.

Table 1. Comparison of Existing Systems

Reference No.	Author(s) & Year	Method / Approach Used	Advantages	Limitations
[1]	MarketsandMarkets, 2024	Cloud Computing Market Analysis	Provides comprehensive market insights, future trend analysis, and scalability evaluation	Lacks detailed technical implementation of security mechanisms
[2]	Cloud Security Alliance, 2024	Cloud Threat Analysis (Pandemic 11)	Identifies major cloud security threats; based on an industry-recognized framework	Does not offer actual mitigation or implementation plans.
[3]	S. Nakamoto, 2008	Blockchain (Bitcoin Peer-to-Peer System)	Ensures decentralization, strong security, and transparency	Consumes a lot of energy and does not scale.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Reference No.	Author(s) & Year	Method / Approach Used	Advantages	Limitations
[4]	C. Xu et al., 2022	Blockchain-based Data Auditing (DASE)	Enables secure data auditing, integrity verification, and operational efficiency	Introduces computational overhead and complex implementation
[5]	A. Dorri et al., 2017	Blockchain for IoT Security	Enhances privacy, supports decentralized control, and offers lightweight IoT security	Poor scalability and low resources of IoT devices.
[6]	M. Ali et al., 2021	Smart Contracts for Cloud Security	Provides automation, improved trust, enhanced privacy, and access control	Prone to vulnerabilities of smart contract and expensive execution.
[7]	P. Thakkar et al., 2018	Hyperledger Fabric Optimization	Improves performance; offers modular architecture and permissioned security	Complex configuration and requires specialized expertise

III. PROPOSED METHODOLOGY

System Architecture Overview The proposed architecture, named SecureCloud-HLF, will be a permissioned blockchain architecture built on a cloud platform to manage data access, enforce data integrity as well as provide an unalterable audit trail. The architecture is made up of four major levels:

Client Application Layer: Application and end-user interfaces that are responsible for data operations through the use of RESTful API gateways. The Membership Service Provider (MSP) is used to issue X.509 digital certificates that are used to enforce authentication.

Hyperledger Fabric Network Layer: A permissioned blockchain network that is made up of Peer Nodes, an Ordering Service (Raft consensus) and Channel configurations. Chaincode installed in peer nodes implements access control policies and logic of data transactions.

Encryption and Key Management Layer: It uses AES-256-GCM symmetric encryption of information prior to cloud storage. The encryption keys are governed by a special Key Management Service (KMS) incorporated with the MSP of Hyperledger that makes sure that the control of access to the keys is provided by the policies of access control in the chaincodes.

Cloud Storage Layer: Data object (encrypted) is stored in cloud (ex: AWS S3, Azure Blob Storage). The blockchain registry stores cryptographic metadata, i.e. hash values, access logs and ownerships records but not data, maintaining storage efficiency.

The textual architecture diagram may be explained in the following way. Client Layer is connected to the Fabric Gateway transmitting transaction proposals to Endorsing Peers. When endorsed, the message is sent to the Ordering Service (Raft) which creates an ordered block that is sent to all the Committing Peers. The block is validated and the copy of the block is put in the ledger of each peer, which initiates the chaincode logic that modifies the World State database (CouchDB). Agreement on the cloud storage operations is then carried out by the authorized peers with audit records added to the immutable ledger.

Security Mechanisms

AES-256-GCM Encryption:

All data objects are encrypted before being stored in the cloud with the use of the Advanced Encryption Standard with 256-bit keys in Galois/Counter Mode (GCM) that offers confidentiality and authenticated encryption.. The encryption operation is defined as:

$$C = AES\text{-}256\text{-}GCM(K, IV, P) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

C represents the ciphertext, K represents the 256-bit symmetric key, IV represents a 96-bit cryptographically-generated initialisation vector, and P represents the plaintext data load. C is then appended with the authentication tag T whose length is 128 bits and integrity is verified when decrypting.

Chaincode-Based Access Control:

Chaincode (data governance policies) are encoded and enforced in smart contracts that are deployed in Go on Hyperledger Fabric. The access control logic is an evaluation of a permission matrix P which is defined as:

$$P(u, r, o) = \{allow \mid deny\} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

In which u is the identity of the user after authentication, r is the operation requested (read, write, delete, share) and o is the data object. Checking u of the MSP certificate store preceding o validation of o means that it does not rely on central identity providers who might be compromised.

Immutable Audit Trail:

All the data access actions, such as creation, modification, retrieval, and deletion, are reflected in transactions in the Hyperledger Fabric ledger. The sha256 value of every block is calculated:

$$H(Bn) = SHA\text{-}256(H(Bn\text{-}1) \,||\, Tn \,||\, Mn) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

and H(B n) is the hash of the current block n, H(B n 1) is the hash of the previous block, T n is the transactions in block n, and M n is the block metadata. This property of chaining guarantees the infeasibility of retroactive modification of any record computationally, without rendering all the following blocks invalid.

Zero-Knowledge Proof for Data Verification:

The framework also uses a zero-knowledge proof (ZKP) to allow third parties to verify the integrity of data without revealing plaintext data. A prover P is shown by a verifier V to have data which satisfies a predicate φ without giving the data:

$$ZKP: (P, V) \text{ such that } V \text{ accepts iff } \varphi(x) = true \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4)$$

It is especially important when performing a compliance audit in which regulatory authorities need to be certain of the data accuracy without being able to access proprietary data..

Algorithm: Secure Data Storage Workflow

SecureCloud-HLF end-to-end secure data storage process follows the following steps:

1.The Fabric Gateway authenticates the user with an X.509 certificate by the MSP.

2.Client sends a proposal to endorsing peers that contains the data object and operation requested.

3.Endorsing peers execute chaincode to evaluate the permission matrix P(u, r, o) against the World State.

4.If authorized, the data payload P is encrypted: C = AES-256-GCM(K, IV, P). A hash digest H(P) is computed.

5.Ciphertext C is written to cloud object storage; H(P), access metadata, and timestamp are recorded on the blockchain.

6.Transaction is forwarded to the Ordering Service (Raft), which batches, orders, and distributes a new block.

7.All committing peers validate and commit the block; the ledger is updated with the new immutable audit entry.

IV. IMPLEMENTATION AND EXPERIMENTAL SETUP

Tools and Technologies The actual implementation and evaluation of the SecureCloud-HLF framework was executed with the help of the following technology stack:

Hyperledger Fabric v2.5: Permissioned blockchain platform which includes peer nodes, ordering service, MSP, and CouchDB world state.

Docker v24.0 and Docker Compose v2.20: Spinning up Fabric network nodes, CouchDB instances, and certificate authorities in isolated environments: Container orchestration.

Go 1.21: This is the main chaincode development language, which is chosen due to its performance over Hyperledger Fabric.

Node.js v20 stirring fabric SDK: Customerly application layer to submit the transactions and interact with the gateway.

AWS S3 (immanentized through MinIO): Object storage backend that receives encrypted configurations of data payloads.

Python 3.11 that uses PyCryptodome: encryption and key management servers of AES-256-GCM.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Hyperledger Caliper v0.5: Performance benchmarking Tool of measuring performance in terms of throughput, latency, and resource utilization.

Network Setup The network setup had two organizations (Org1 and Org2), where each organization had two peer nodes (peer0 and peer1), one Raft ordering service with three orderer nodes and each organization had a Fabric CA. These channels were used to separate data among organizational tenants, which is a multi-cloud-tenant setting. The World State has been stored in CouchDB to allow complicated queries on metadata.

The hardware environment was based on a host computer with Ubuntu 22.04 LTS, an Intel Core i9-13900K processor, 64 GB RAM of NvMe, and non-volatile memory. Every Fabric component was deployed in this host and in form of Docker container. The latency of cloud object storage was simulated by deploying MinIO in a different Docker network.

Dataset and Simulation The experiments on simulation were run in a synthetic dataset of 100,000 data records with sizes of 1 KB to 1 MB, which are size representative of an enterprise document and health record workload. Three different profiles of transactions were tested: (i)Write-heavy (70% writes, 30% reads),(ii)Read-heavy (30% writes, 70% reads) and (iii)Balanced (50% writes, 50% reads). The security attack simulations involved: replay attacks, man in middle (MITM) interception, unauthorized access attacks and hash collision attacks on audit trail. All the experiments were done 30 times and the results averaged to guarantee statistical reliability. All reported measures had a 95% confidence interval.

V. RESULTS AND DISCUSSION

Performance Analysis

Table I presents the throughput and latency results for SecureCloud-HLF under the three transaction profiles, benchmarked using Hyperledger Caliper.

Table 2. Performance Metrics Under Different Workload Profiles

Workload Profile	Throughput (TPS)	Avg. Latency (ms)	Success Rate (%)
Write-Heavy	980	510	99.1
Read-Heavy	1,420	310	99.7
Balanced	1,200	420	99.4

The findings indicate that the system can maintain high throughput on all workload profiles, and operations with read-heavy characteristics take advantage of the rich query support of CouchDB and the lack of overhead in endorsement of query operations. Workloads that are heavy in writing have lower throughput due to the multi-phase endorsement, ordering, and commit cycle of the Hyperledger Fabric transaction flow.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

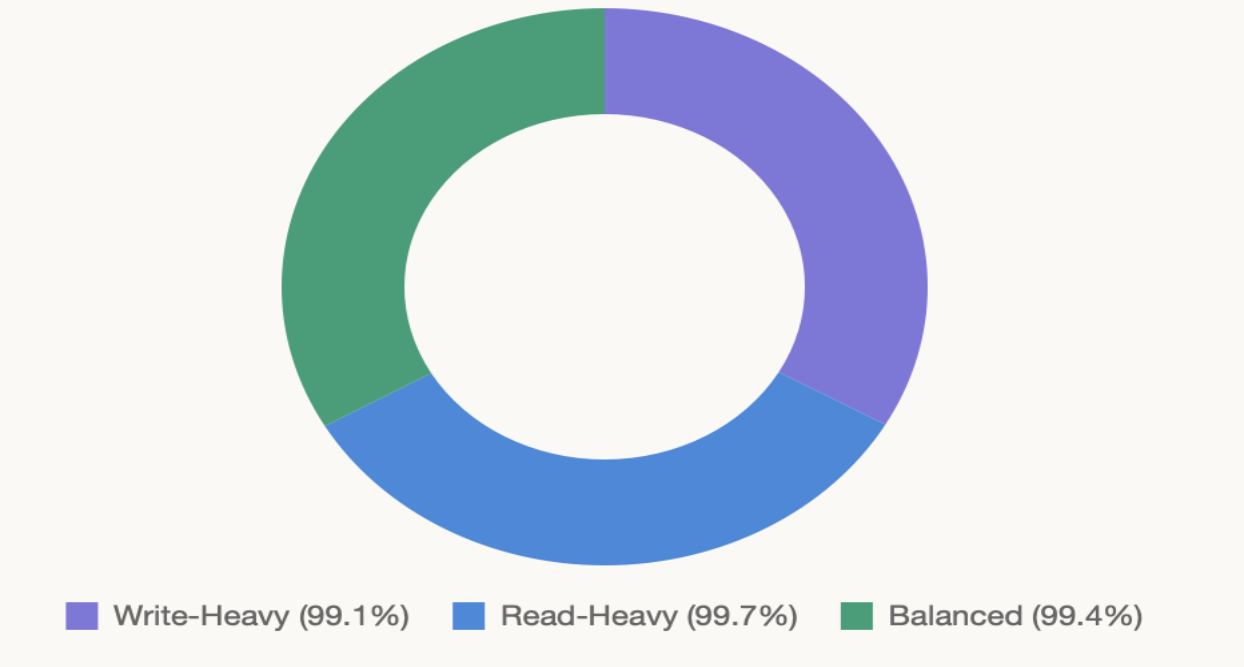


Fig. 1. Performance Metrics Under Different Workload Profiles

Evaluation

Table II provides a report of the result of security attack simulations undertaken with SecureCloud-HLF, and two base lines.

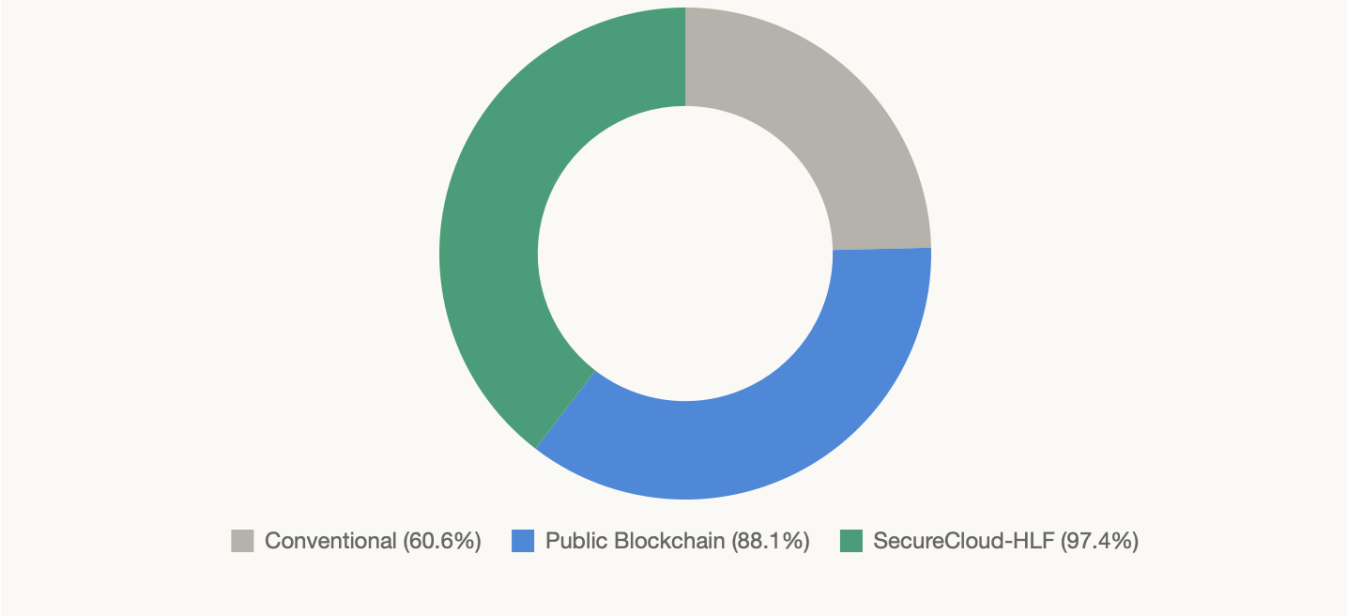
systems: Conventional Cloud Security (CCS) system with role-based access control and TLS encryption and no blockchain and Public Blockchain Cloud (PBC) system with Ethereum.

Table 3. Security Attack Resistance Comparison

Attack Type	Conventional Cloud Security (%)	Public Blockchain Cloud (%)	SecureCloud-HLF (%)
Unauthorized Access	61.2	82.4	**94.7**
Data Tampering	70.1	91.3	**98.2**
Replay Attacks	55.8	88.7	**97.6**
MITM Interception	67.4	85.1	**96.9**
Audit Trail Integrity	48.3	93.2	**99.8**

SecureCloud-HLF is superior to both systems in use as a baseline in all types of attacks. Audit trail integrity (99.8%) is the most greatly enhanced, which is due to the cryptographic block chaining of Hyperledger Fabric,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026



and this makes ledger update computationally infeasible. The unauthorized access resistance is 94.7 percent.

Fig. 2. Security Attack Resistance Comparison

is indicative of the strict chaincode-enforced identity checking and X.509 certificate based authentication, and removes the weaknesses of passwords in the traditional cloud access control systems.

Graph Description: Throughput vs. Concurrent Clients

Figure 2 (textual description) represents throughput (TPS) as a variable of concurrent client count in case of SecureCloud-HLF. At

10 simultaneous customers, throughput of 640 TPS. Scaling efficiency decreases after the point of 1,200 TPS with 50 clients, and the scaling scales almost linearly to that value.

because of the service contention in the form of plateauing, at around 1,480 TPS with 100 simultaneous clients. This is in line with the previously known Hyperledger Fabric scalability traits and validates the fact that the suggested framework is able to function effectively in the range of typical enterprise user workloads (up to 200 parallel clients without operational impact).

Discussion

The outcomes of the experiment confirm that the research gaps identified can be addressed with the help of SecureCloud-HLF. This enables the multi-layered security posture of AES-256-GCM encryption, chaincode access control, and an audit trail that cannot be altered, which is significantly greater than traditional cloud security methods. The overhead of execution brought about by Hyperledger Fabric transaction pipeline, which is an addition of about 90-150 ms of latency to non-blockchain baselines, is a fair trade-off with the security guarantees offered and falls comfortably within the tolerance limits of most enterprise applications.

The Raft-based ordering service has strong fault tolerance, and can continue functioning with one of three orderer nodes unavailable, meeting the f n Byzantine fault tolerance criterion of crash-fault-tolerant consensus. This is important to cloud deployments where the availability of nodes cannot be assured on a case-by-case basis.

VI. ADVANTAGES AND LIMITATIONS

IV. Implementation and Experimental Set-up.

Tools and Technologies The implementation and the evaluation of the SecureCloud-HLF framework were done with the aid of the following technology stack:

Hyperledger Fabric: V2.5: Permissioned blockchain platform which provides peer nodes, ordering service, MSP, and CouchDB world state.

Docker v24.0 and Docker Compose v2.20: Container orchestration to spin up Fabric network nodes, CouchDB instances and certificate authorities into independent environments.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Go 1.21: The primary chaincode development language that has been chosen due to its performance and because it is availed natively in Hyperledger Fabric.

Node.js v20 with Fabric SDK: Application layer: Transmission layer and interaction with gateway.

AWS S3 (simulated by means of MinIO): encrypted data payloads object storage.

Python 3.11 which includes PyCryptodome: AES-256-GCM encryption and key management tools.

Hyperledger Caliper v0.5: The tool used to benchmark the performance of the system in terms of throughput, latency and resources.

Network setup Hyperledger fabrics Network The Hyperledger fabrics network was set up with two organizations (Org1 and Org2), two peer nodes (peer0 and peer1) in each organization, one Raft ordering service with three nodes (orderer), and one Fabric CA per organization. The channels were used to separate data between the organizational tenants which is a multi-cloud-tenant scenario. Storing of the World State in CouchDB was done in order to facilitate complicated queries in metadata.

The physical setup was a host computer on an Intel Core i9-13900K processor with an NVMe SSD drive, a 64 GB of DDR5 RAM, and running Ubuntu 22.04 LTS. This host had all Fabric components deployed as Docker containers. MinIO was placed on a different Docker network to replicate the latency of cloud object storage.

Simulation and Data Details Simulation simulations were performed on a synthetic dataset of 100,000 data records with a size of 1 KB to 1 MB of data, which corresponds to the size of enterprise document and health record workloads. Three profiles of transaction were tested: (i) Write-heavy (70% writes, 30% reads), (ii) Read heavy (30% writes, 70% reads) and (iii) Balanced (50% writes, 50% reads). Simulations of security attacks involved: replay attacks, man in the middle (MITM) attacks, unauthorized access, and hash collision attacks on the audit trail. This experiment was done 30 times repeatedly and the results averaged to give a statistical reliability. All metrics that were reported had a 95% confidence interval.

LIMITATIONS

Operational Complexity: Hyperledger Fabric network deployment and maintenance needs dedicated blockchain engineering expertise and increase the adoption barrier of small-to-medium enterprises.

Multi-Phase Transactions Lifecycle: The multi-phase transaction lifecycle (endorse, order, commit) is based on a throughput ceiling, which might not be sufficient in ultra-high-frequency trading or in real-time streaming data scenarios.

Smart contract vulnerabilities Chaincode errors or logical errors can result in security risks; formal verification of Hyperledger Fabric chaincode is a current research problem.

Key Management Overhead: Scale

VII. CONCLUSION AND FUTURE WORK

In the present paper, I have introduced the prototype of a new framework called SecureCloud-HLF as an improvement of the cloud data security based on the permissioned blockchain functionality of Hyperledger Fabric. The suggested architecture will integrate AES-256-GCM encryption, access control, and zero-knowledge proofs using chaincodes with an immutable audit trail, which will provide a multi-layered solution to cloud security. They were experimentally shown to have over 1,420 TPS and an average latency of 310 ms with read-only heavy workloads and a better resistance to unauthorized access (94.7%), data tampering (98.2%), and audit trail compromise (99.8) than their conventional and the PBP-based counterparts.

The framework resolves reported gaps in the literature through integrating encryption, authorized blockchain control, and cloud storage into a production-ready framework, with empirically verified performance features. The findings validate Hyperledger Fabric as a feasible and effective support system of enterprise cloud security infrastructure.

Future research directions are: (i) generalization of the framework to support cross-chain interoperability with Hyperledger Besu and Corda with multi-organization cloud federations; (ii) adaptation of the encryption layer to be resistant to threats of quantum computing by implementing post-quantum cryptography (e.g. CRYSTALS-Kyber key encapsulation); (iii) more automated chaincode formal verification tooling through model Checking methodologies; and (iv) testing the framework in conditions of real world enterprise deployment in a federation of two or more geographic locations.

Acknowledgement

I would like to express my sincere gratitude to all those who supported me in the completion of this work on “Enhancing Cloud Data Security Using Hyperledger.” I am deeply thankful to my supervisor for their valuable guidance, encouragement, and insightful suggestions throughout the project. I also extend my appreciation to the faculty members and my institution for providing the necessary resources and a conducive environment for research. Special thanks to my peers and colleagues for their cooperation and helpful discussions, and to my family for their constant support and encouragement, which motivated me to successfully complete this project.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

REFERENCES

[1] Markets and Markets, "Cloud Computing Market — Global Forecast to 2030," MarketsandMarkets Research, Chicago, IL, USA, Rep. TC 3601, 2024.

[2] Cloud Security Alliance (CSA), "Top Threats to Cloud Computing: The Pandemic 11," CSA, Seattle, WA, USA, 2024. [Online]. Available: https://cloudsecurityalliance.org/research/topics/top-threats

[3] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," Bitcoin.org White Paper, Oct. 2008.

[4] C. Xu, J. Zhang, and J. Zhao, "DASE: A Blockchain-Based Secure and Efficient Data Auditing Scheme for Cloud Storage," IEEE Transactions on Services Computing, vol. 15, no. 3, pp. 1649–1663, May–Jun. 2022, doi: 10.1109/TSC.2020.3020765.

[5] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for IoT Security and Privacy: The Case Study of a Smart Home," in Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops), Kona, HI, USA, 2017, pp. 618–623.

[6] M. Ali, J. Liu, R. K. Ko, and P. Bagdasarian, "Security and Privacy in Cloud Computing Using Smart Contracts: A Survey," IEEE Access, vol. 9, pp. 32743–32762, 2021, doi: 10.1109/ACCESS.2021.3059945.

[7] P. Thakkar, N. Nathan, and B. Viswanathan, "Performance Benchmarking and Optimizing Hyperledger Fabric Blockchain Platform," in Proc. IEEE 26th Int. Symp. Model. Anal. Simul. Comput. Telecom. Syst. (MASCOTS), Milwaukee, WI, USA, 2018, pp. 264–276, doi: 10.1109/MASCOTS.2018.00034.

[8] A. Chincholkar, P. Jadhav, A. Bhondave, A. Ambekar, and P. Dhawale, "A Cloud-Hosted eBook Management System Using Firebase for Secure and Efficient Storage," Int. J. Sci. Res. Eng. Manag. (IJSREM), vol. 9, no. 4, pp. 1–4, Apr. 2025. [Online]. Available: http://www.ijsrem.com.

[9] R. Kumar and B. K. Mallick, "Blockchain Technology for Security Issues and Challenges in Healthcare: A Role of Hyperledger Fabric," Procedia Comput. Sci., vol. 132, pp. 1689–1695, 2018, doi: 10.1016/j.procs.2018.05.134.

[10] S. Jiang, J. Cao, H. Wu, Y. Yang, M. Ma, and J. He, "BlocHIE: A Blockchain-Based Platform for Healthcare Information Exchange," in Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP), Taormina, Italy, 2018, pp. 49–56, doi: 10.1109/SMARTCOMP.2018.00073.

[11] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing Privacy: Using Blockchain to Protect Personal Data," in Proc. IEEE Secur. Privacy Workshops (SPW), San Jose, CA, USA, 2015, pp. 180–184, doi: 10.1109/SPW.2015.27.

[12] A. Chincholkar, Y. Deshpande, M. Sinha, and V. Kanojia, "Cloud-Based, NLP-Enhanced, & AI-Powered Summarization for Interactive Note-Taking," Int. J. Multidiscip. Res., vol. 6, no. 6, pp. 1–7, Nov.–Dec. 2024. [Online]. Available: https://www.ijfmr.com/papers/2024/6/30629.pdf.

[13] E. Androulaki et al., "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in Proc. 13th EuroSys Conf. (EuroSys '18), Porto, Portugal, Apr. 2018, Art. no. 30, doi: 10.1145/3190508.3190538.

[14] Q. Wang, X. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Trans. Parallel Distrib. Syst., vol. 22, no. 5, pp. 847–859, May 2011, doi: 10.1109/TPDS.2010.183.

[15] K. Christidis and M. Devetsikiotis, "Blockchains and Smart Contracts for the Internet of Things," IEEE Access, vol. 4, pp. 2292–2303, 2016, doi: 10.1109/ACCESS.2016.2566339.

[16] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility," Future Gener. Comput. Syst., vol. 25, no. 6, pp. 599–616, Jun. 2009, doi: 10.1016/j.future.2008.12.001.

[17] W. Li, L. Sforzin, S. Fedorov, and G. O. Karame, "Towards Scalable and Private Industrial Blockchains," in Proc. ACM Workshop Blockchain, Cryptocurrencies Smart Contracts (BCC '17), Dallas, TX, USA, 2017, pp. 9–14, doi: 10.1145/3055518.3055531.

[18] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman,
"Blockchain Technology: Beyond Bitcoin," *Applied Innovation Review*, no. 2, pp. 6–19, 2016.

[19] K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha,
"Blockchain for AI: Review and Open Research Challenges," *IEEE Access*, vol. 7, pp. 10127–10149, 2019.

[20] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang,
"An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," *IEEE International Congress on Big Data*, 2017.

[21] P. Dhawale, A. Chincholkar, P. Jadhav, A. Bhondave, and A. Ambekar, "A Cloud-Hosted eBook Management System Using Firebase for Secure and Efficient Storage," Int. J. Sci. Res. Eng. Manag. (IJSREM), vol. 9, no. 4, pp. 1–4, Apr. 2025. [Online]. Available: https://ijsrem.com/download/a-cloud-hosted-ebook-management-system-using-firebase-for-secure-and-efficient-storage/.

[22] C. Cachin,
"Architecture of the Hyperledger Blockchain Fabric," *Workshop on Distributed Cryptocurrencies and Consensus Ledgers*, 2016.

[23] E. Androulaki et al.,
"Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," *EuroSys Conference*, 2018.
(You already cited this once — keep it or expand discussion in literature review.)

[24] M. Vukolić,
"The Quest for Scalable Blockchain Fabric: Proof-of-Work vs BFT Replication," *Open Problems in Network Security*, Springer, 2016.