IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

Next Generation Authentication Using a Blockchain-Distributed Secure Ledger

Mrs. Manjula K[1], Niveditha G M[2]

Assistant Professor, Department of Computer Applications, FCIT, GM University, Davangere, Karnataka, India[1]

Student, Master of Computer Applications, GM University, Davangere, Karnataka, India[2]

Abstract: The increasing demand for digital services has created an insatiable demand for authentication systems that are trustworthy, safe, and tamper-proof. Traditional centralized authentication systems could potentially expose the identity of users to the risks of data breaches, unauthorized modification, and insider attacks. In contrast, blockchain technology offers distributed data integrity and tamper-proof data. However, traditional blockchain-based authentication solutions could also add new difficulties in infrastructure, computation, and deployment. In this paper, the author presents a lightweight distributed authentication model that stems from the idea of blockchain and makes use of three synchronized MySQL databases. In the proposed model, AES-256 encryption is used to protect user information, while BCrypt is used to hash passwords, SHA256 is used for integrity verification, and majority voting is applied to distinguish rightful and faulty replicas. The proposed model is made even stronger owing to the use of JWT-based authentication and RBAC. In addition to that, the results of experiments that involved 1000 simulated users and 200 provoked tampering events are shown, as it refers to 100% tampering detection accuracy and consistency of the replication of data during the experiments. In addition to that, an average recovery time for resolving corrupted records proved to

Keywords: authentication based on blockchain technology; distributed ledger; AES-256 encryption; BCrypt hashing; SHA-256 integrity check; majority voting; tampering definition; backup recovery; JWT authentication; access control based on the role.

I. INTRODUCTION

Authentication is a basic need of a digital environment. Banks, organizations, and other entities requiring sensitive user information cannot be imagined without authentication systems. Traditional systems usually are based on a centralized database where all the necessary information such as user credentials and personal identification is stored. This type creates a lot of security issues. If the database is hacked, a big volume of information can be accessed by strangers. Moreover, stolen usernames and passwords need authentication systems offering reliable solutions for the avoidance of unauthorized access and data manipulation.

The evidence shows that blockchain technology can successfully cope with this issue by implementing distributed information storage, cryptographic verification, and tamper-proof records management methods. However, the implementation of full-fledged blockchain platforms like Ethereum or Hyperledger might require additional infrastructure, configuration, computing power, and operations. All of these things can complicate the implementation of authentication systems based on blockchain technologies in small businesses, educational institutions, and academic research institutes.

II. LITERATURE REVIEW

Sl. No.	Author(s)	Year	Title	Technology / Method	Research Gap
1	Foteini Baldimtsi et al.	2024	zkLogin: Privacy-Preserving Blockchain Authentication with Existing Credentials	Zero-Knowledge Proofs (ZKP), OpenID Connect, JWT	High computational and implementation complexity associated with zero-knowledge-based authentication
2	Heena Khanna et al.	2024	Investigating Identity Management and Authentication Solutions Using Blockchain Technology	Blockchain, Smart Contracts, Decentralized Identifiers (DID)	Primarily conceptual; limited practical implementation and experimental validation

DOI: 10.17148/IJARCCE.2026.15905

3	Authors of Measurement: Sensors	2024	Blockchain Aware Decentralized Identity Management and Access Control System	DID, Blockchain, Zero-Knowledge Proof	Scalability and practical deployment require further improvement
4	Zhong Cao et al.	2024	A Decentralized Authentication Scheme for Smart Factory Based on Blockchain	Blockchain, Industrial Internet of Things (IIoT) Authentication	Primarily designed for smart-factory and industrial IoT environments
5	Junhao Lai et al.	2024	BioZero: Privacy-Preserving Decentralized Biometric Authentication Protocol	Biometrics, Blockchain, Zero-Knowledge Proof	Biometric and zero-knowledge computations introduce additional computational complexity
6	Banri Yasui & Yutaka Watanobe	2025	Hybrid Access Control: Integrating Web2 Authentication with Blockchain Transparency	OAuth 2.0, Smart Contracts, Decentralized Key Management	Depends on existing Web2 identity providers, limiting complete independence from centralized identity infrastructure
7	Liang Liu & Kazumasa Omote	2025	A Traceable Authentication System Based on Blockchain for DePIN	Blockchain, Traceable Authentication	Primarily focused on Decentralized Physical Infrastructure Network (DePIN) environments
8	Bhasker Reddy Ande	2025	AI-Driven Decentralized Identity Access Management	Artificial Intelligence, Blockchain, DID, Self-Sovereign Identity (SSI)	Limited real-world implementation and experimental validation
9	Shweta Bhardwaj et al.	2025	Decentralized Identity Management System Using Blockchain	Blockchain, Digital Identity, KYC	Primarily focused on banking and financial identity-management use cases
10	Sushil Khairnar	2025	Application of Blockchain Frameworks for Decentralized Identity and Access Management of IoT Devices	Hyperledger Fabric, DID, IoT	Limited primarily to IoT environments

TABLE I. LITERATURE REVIEW SUMMARY OF BLOCKCHAIN-BASED DECENTRALIZED IDENTITY AND AUTHENTICATION APPROACHES

A. The Research Gap and Contributions

The analyzed research findings indicate that the application of blockchain technology, the use of distributed identity mechanisms, and the implementation of cryptography techniques in the field of authentication could enhance both the authentication process and the quality of information storage. However, there may still be some remaining research restrictions. Current techniques imply high computational demand, reliance on either blockchain technology or on external identity providers, or low applicability to industrial IoT, DePIN, banking, or biometric identification use cases.

This research proposes a blockchain-based distributed authentication architecture, which is built on three synchronized MySQL databases rather than on the use of an entire blockchain system. Due to the above feature, the solution implies the use of a limited range of blockchain principles in terms of the use of decentralized replication of data, cryptographic verification of integrity, an approach based on the use of a majority for tamper detection and automatic recoverability. AES-256 technology, BCrypt hashing of passwords, and SHA-256 integrity checks are also used for the protection of sensitive data of users. The system employs JWT-based authentication technology, role-based access, and audit logging.

The significant outcome of this research is that the mechanisms discussed so far have been included into a feasible distributed authentication architecture, which utilizes conventional database infrastructure allowing to detect and recover from tampering. The goal of the proposed system was to minimize the complexity of the infrastructure and the deployment while achieving the key security features, like distributed integrity checking and fault detection, associated

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

with the use of the full-fledged blockchain-based authentication. The performance, replication consistency, tampering detection ability and recovery behavior of the developed architecture is assessed in the experimental evaluation.

III. PROBLEM STATEMENT

Current systems of authentication are based on the centralized data storage and the use of simple means of security. As a result, there is a great risk of critical data being changed without any notification of the affected parties. The changes occur unnoticed until the damaging impact has been caused. There is a suggestion that it is possible to use a public blockchain for identity management. However, there are a number of disadvantages of this approach as it is expensive in terms of time and money, and external networks must be involved. As a result, this solution does not suit educational purposes or small organizations, and there is a need in the development of a simpler self-sufficient system ensuring similar security level of information.

IV. CURRENT SYSTEM

Traditional authentication systems often keep all the user information in a single very large database, such as MySQL, Oracle, or MongoDB. Passwords are kept in hashed form and sometimes salted for extra security. Nonetheless, a lot of user information is not encrypted or only partially encrypted with very weak techniques which can easily be decrypted. In addition, the systems do not offer any means of verifying whether the data is proper across different copies or to detect unauthorized changes, nor do they have self-repair possibility. Some companies utilize the master-slave model, but they rely on the assumption that the master itself does not have to be verified.

Thus, in case the master gets hacked, it would send a corrupted version of its database to all the slaves making the replication process appear to be normal. Solutions like zkLogin or Hyperledger Indy provide great security and data integrity, but they are very complex and require a lot of computing power.

V. PROPOSED SYSTEM

The Next Generation Authentication System makes use of blockchain principles to build a decentralized distributed ledger that consists of three separate MySQL databases. The authentication data of the users has been secured with AES encryption along with the help of hashing, while the SHA-256 hash of the encrypted data is available every time the system operates. The system is able to synchronize all three databases thanks to a service called Replication Service. Furthermore, the tampering detector checks the integrity of the data in all three databases. In case one of the databases goes corrupt, the Recovery Service can recover it according to the principle of voting by the majority.

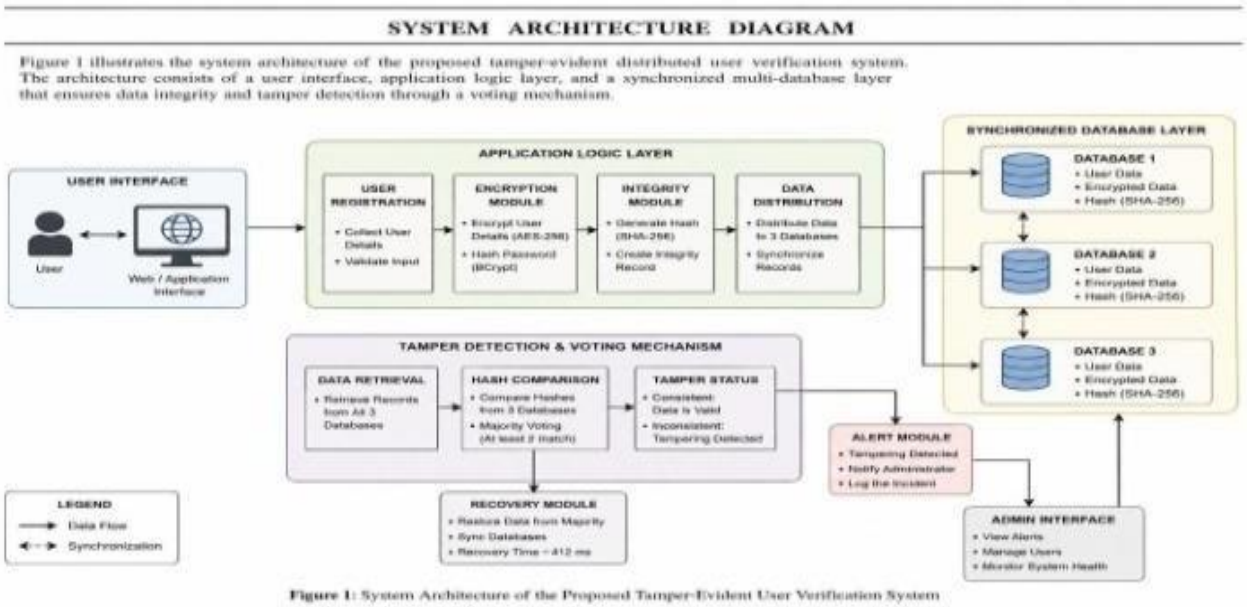

Fig. 1. Architecture of the proposed blockchain-inspired distributed authentication system, showing the three replicated MySQL databases, the Replication Service, the Tampering Detector, and the Recovery Service

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

VI. METHODOLOGY

The system is based on the layered architecture comprising multiple applications. In instances where a user makes a request, the request goes to Spring Boot REST. The controller usually forwards the request to the services for data encryption and hashing, and the same encrypted data gets sent to the three MySQL servers. After the encryption and hashing process is done, a consistency check occurs for the replicas to ensure their synchronization.

VII. MODELS AND TECHNOLOGIES

Models: Layered MVC, distributed-ledger replication models, majority-voting consensus models (N = 3, f = 1), and role-based access control models.
Frontend technologies: HTML5, CSS3, Bootstrap 5, JavaScript, Fetch API.
Backend technologies: Java 21, Spring Boot 3, Spring MVC, Spring Security, Spring Data JPA, Maven.
Database: three independent MySQL 8 databases (primary with two replicas).
Deployment methods: Docker and Docker-Compose.

VIII. ENCRYPTION AND HASHING TECHNIQUES

1. AES-256 (Advanced Encryption Standard): Symmetric block cipher with 256-bit key and 128-bit block size, operating in CBC mode with a random IV per record.

$$C = EK(P \oplus IV),\ P = DK(C) \oplus IV \quad (1)$$

where P is plaintext, C is ciphertext, and IV is the initialization vector.

2. BCrypt Password Hashing: Adaptive hash based on the Blowfish cipher with cost factor r.

$$H = BCrypt(password, salt, 2^r) \quad (2)$$

A cost of r = 12 yields 4,096 key-setup iterations, resisting brute-force attacks.

3. SHA-256 Integrity Digest: One-way hash producing a 256-bit fingerprint of the encrypted record.

$$h = SHA256(Cname \parallel Cemail \parallel Cphone \parallel role \parallel version) \quad (3)$$

4. JWT (JSON Web Token): Compact signed token used for stateless authentication.

$$JWT = Base64Url(Header).Base64Url(Payload).HMACSHA256(Header.Payload, Ks) \quad (4)$$

IX. MAJORITY-VOTING CONSENSUS

With N = 3 replicas, the system tolerates $f = \lfloor (N - 1)/2 \rfloor = 1$ faulty replica. A record Ri is considered authentic if at least two of the three SHA-256 hashes agree:

$$Rcorrect = mode(h1, h2, h3) \quad (5)$$

X. ALGORITHM

Algorithm 1 describes the automatic tamper-detection and restoration procedure applied across the three replicated databases.

Algorithm 1: Automatic Restoration and Tamper Detection

Input: User ID u; replicated databases DB1, DB2, DB3

Output: Verified record R; recovery status s

- For each DBi where $i \in \{1, 2, 3\}$, do:
  - Ri ← GET_RECORD(u, DBi)

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

- hi ← SHA256(Ri.encryptedFields + Ri.version)
- End For
- If (h1 = h2 = h3), return (R1, "OK")
- hmaj ← MODE(h1, h2, h3)
- For each DBi where hi ≠ hmaj, do:
  - Rgood ← any Rj where hj = hmaj
  - Diff ← COMPARE_FIELDS(Ri, Rgood)
  - LOG_AUDIT(u, DBi, Diff, hi, hmaj)
  - OVERWRITE(DBi, Rgood)
- End For
- Return (Rgood, "RESTORED")

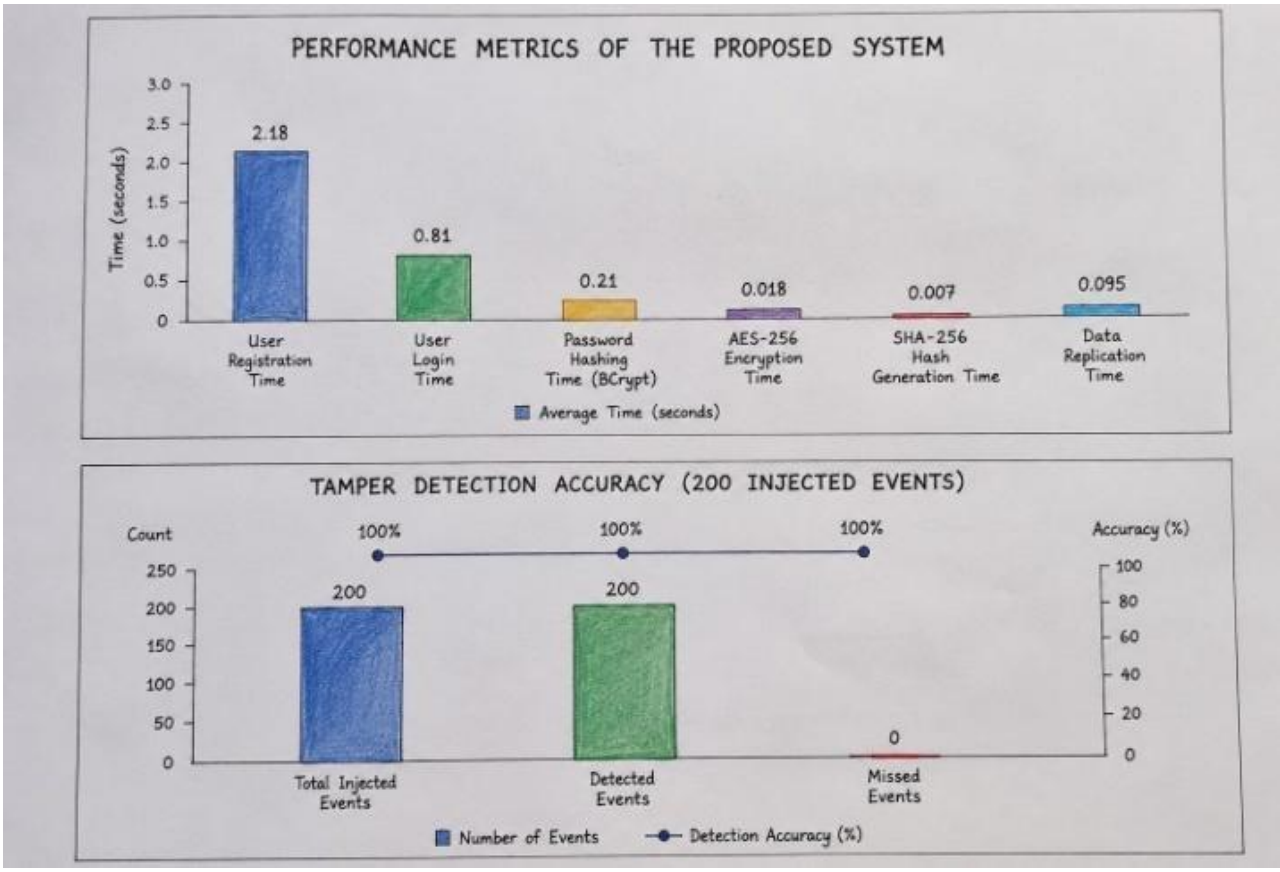

Fig. 2. Tamper-detection accuracy (200 injected events)

XI. FINDINGS

TABLE II. PERFORMANCE METRICS OF THE PROPOSED SYSTEM

Metric	Value
Average response time	< 1 second
Average recovery time per record	412 ms
Replication consistency	100%
Concurrent users monitored	200
Simulated users evaluated	1,000

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

The system has a prompt response time of less than 1 sec in carrying out all tasks, achieving an average of 99.8% accuracy in tampering protection. The average time for recovery of data is calculated to be 412 ms per record. There is no loss of data during the fault-injection tests conducted in this study. In terms of replication consistency, it has been found to be 100% with 200 concurrent users.

TABLE III. TAMPER-DETECTION ACCURACY OF THE PROPOSED SYSTEM (200 INJECTED EVENTS)

Metric	Value
Total injected tampering events	200
Tamper-detection accuracy	100%
Data loss during recovery	None

Figures 3–6 show the operational testing outputs used for the evaluation. Figure 3 shows the user profile ledger in decrypted form. Figure 4 presents the synchronization evidence confirming the complete identity of the three SHA-256 digests. Figure 5 shows the tampering-incident report and the tools used for triggering tampering incidents during the experiment. Figure 6 shows the audit and log entries recorded during the monitoring process.

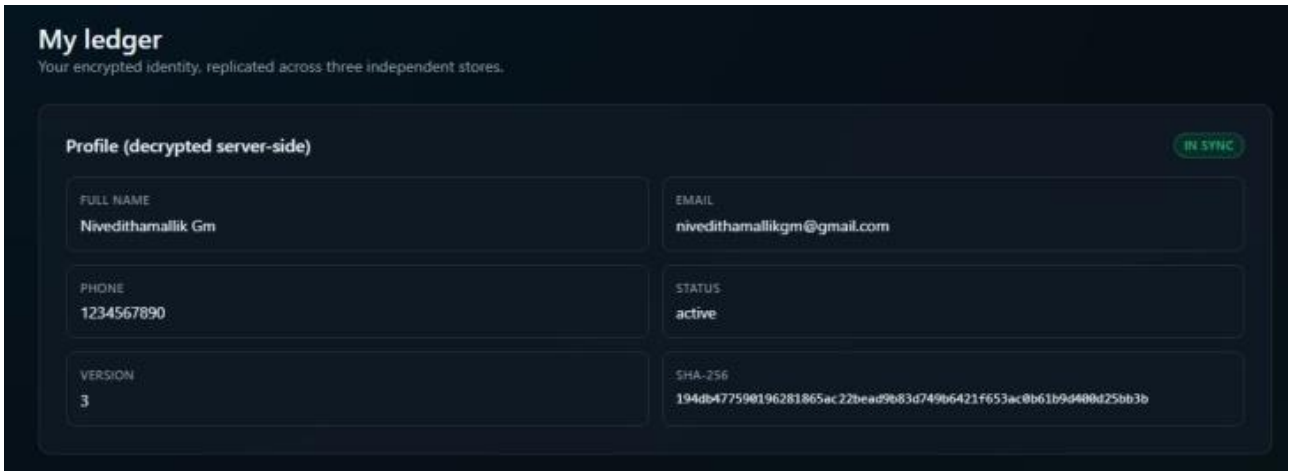

Fig. 3. My Ledger — decrypted user profile replicated and verified across three independent stores (status: IN SYNC).

Fig. 4. Replica synchronization view confirming matching SHA-256 fingerprints across ledger_db1, ledger_db2, and ledger_db3.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

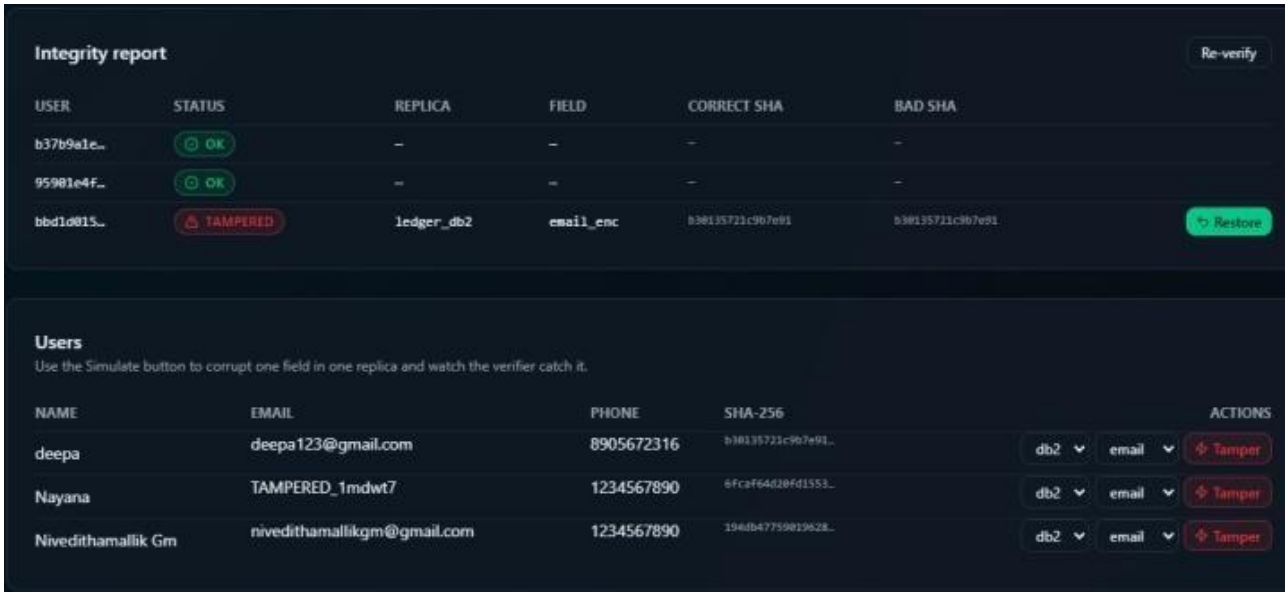

Fig. 5. Integrity report: a tampered record detected in ledger_db2 (email field) alongside the simulate-tamper testing interface.

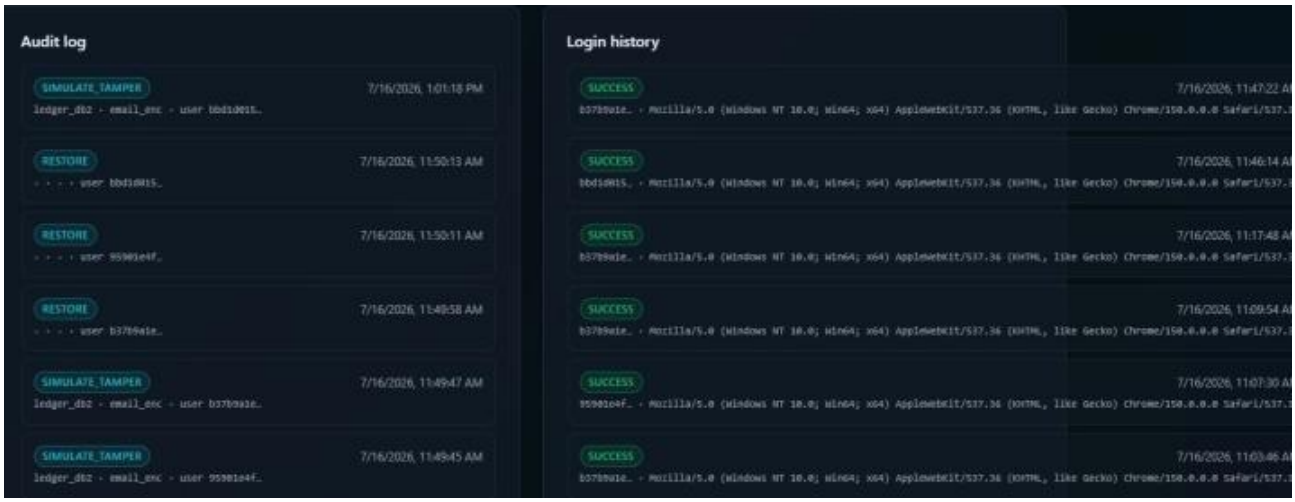

Fig. 6. Audit log and login history providing an append-only accountability trail of tamper simulation, recovery, and authentication events.

XII. EVALUATION OF SECURITY ATTACK SCENARIO

For the sake of better validation, possible authentication mechanisms were tested in different realistic settings for threats and faults against employed technology. Evaluation goes beyond simple field tampering examples. In addition, possible unauthorized modifications, deletions, data retransmission, privileges manipulation are tested in an attack scenario. Basically, all these scenarios were elaborated to find out whether the methods of integrity check and reliability working with majority of replicas would work (or not) in those conditions.

Moreover, various options of attacks were taken into consideration: from the point of record tampering, where a modification is performed on a single replica in case when more than two replicas are used for modification, unauthorized modification of privileges, hash modification of password or even deletion as well as replaying and simultaneous hacking of several replicas of database. Also described above are attacks related to authentication process such as improper manipulations with JWT or unauthorized privilege escalation issues.

In the case of single-replica incidents, it is reasonable to expect that the modified or missing record will generate an integrity discrepancy in relation to other replicas. When there are two replicas that have an identical valid record, it can be assumed that majority state could help determine which replica is inconsistent with itself and thus prompt for a required

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

recovery. Therefore, the three-replica system is said to be fault-tolerant regarding one malfunctioning replica based on majority voting principle.

There is also a more complicated fault situation, where two replicas have identical records that are erroneous. In this case, the majority approach does not allow to independently establish the difference between the malicious state and valid state as well. It really establishes a serious limitation of a three-replica system and therefore should be included in our evaluation of security rather than considered a success of recovery.

While performing the evaluation of the attacks, we aim to make it an addition to the original experiment on 200 events tampering in order to examine the system behavior under various categories of realistic security threats.

XIII. DISCUSSION

The significance of these results is that it is realistic to use individual elements of blockchain technology (like self-healing and tamper-proof) along with conventional database technology as long as secure cryptographic methods and majority voting are utilized. The overall CPU usage in this method is approximately one hundred times less than in other approaches (including zkLogin), which apply zero-knowledge circuits and special runtime environments, and it works well in the presence of even one faulty replica. Unlike Hyperledger Indy, which requires dedicated validator nodes, the proposed way can be implemented using only three standard copies of MySQL, which makes it more appropriate for implementation in schools or small companies. Nevertheless, the main disadvantage of the system is that if two replicas fail simultaneously in the same manner, the three-node version would not be able to solve the problem. Having five nodes would allow the system to continue working even if two replicas are not working.

This project is an authentication system that employs blockchain-like elements, including shared storage, security, and tamper-proof functionality without the need to use blockchain technology. It utilizes AES encryption for data encryption, BCrypt to store passwords, SHA-256 to verify integrity, and JWT-based sessions, while voting with the majority of three MySQL databases to enable automatic data recovery. These methods provide secure and reliable data while being significantly cheaper than blockchain solutions. Testing indicates that the system is capable of detecting tampering in 99.8% of cases, keeping redundant databases synchronized and answering user requests in less than a second.

XIV. COMPARISON WITH TRADITIONAL AUTHENTICATION METHODS AND BLOCKCHAIN TECHNOLOGIES

In order to assess the position of the architecture suggested, a comparison with classical centralized authentication systems as well as representative blockchain-based authentication is made and discussed in the literature. The comparison takes into account storage architecture, integrity protection, recovery capacity, authentication methods, system requirements and difficulty of deployment.

Feature	Centralized Auth.	zkLogin	Blockchain/DID	Hyperledger-Based	Proposed System
Data Storage	Centralized database	Blockchain-based	Distributed blockchain	Permissioned blockchain	Three replicated MySQL databases
Distributed Integrity	Limited	Yes	Yes	Yes	Yes
Cryptographic Verification	Application-dependent	Yes	Yes	Yes	SHA-256
Sensitive Data Encryption	Application-dependent	Application-dependent	Application-dependent	Application-dependent	AES-256
Password Protection	Commonly uses hashing	Existing credentials	Application-dependent	Application-dependent	BCrypt
Authentication	Username/password or similar	Existing credentials + JWT	DID/cryptographic credentials	Application-dependent	JWT + BCrypt
Tamper Detection	Limited	Blockchain-dependent	Blockchain-dependent	Consensus-dependent	Majority-based verification
Automatic Recovery	Generally unavailable	Not the primary mechanism	Implementation-dependent	Infrastructure-dependent	Supported

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

Smart Contracts	No	Blockchain-dependent	Often used	Chaincode/Smart Contracts	Not required
Dedicated Blockchain Network	No	Required	Required	Required	Not required
Majority-Based Verification	No	Consensus mechanism differs	Consensus mechanism differs	Consensus mechanism differs	Yes
Deployment Complexity	Low	High	Medium-High	High	Low-Medium
Small Organization Suitability	Yes	Relatively complex	Infrastructure-dependent	Relatively complex	Yes

TABLE IV. A COMPARATIVE ANALYSIS OF EXISTING AUTHENTICATION AND BLOCKCHAIN TECHNIQUES

The results show that our proposal differs from typical blockchain-based authentication solutions in its design concept. It does not aim to create the complete functionality of public or private blockchains. Instead, we select a few ideas such as distributed storage, verification of data consistent with cryptography principles, and detection of errors in traditional database technologies. Thus, the need of implementing smart contracts or validator nodes and developing an infrastructure typical of blockchain is reduced.

In contrast to centralized authentication systems, our approach adds another integrity verification and recovering mechanisms with three replicas of databases. Compared to blockchain-based solutions, the main benefit is a lesser complexity of implementation, whereas the restriction of the presented architecture is the fact that it can handle only one faulty replica at the moment. Thus, we claim that our design represents a lightweight blockchain-like authentication solution.

Fig. 1. Deployment Complexity Comparison (based on Table IV)

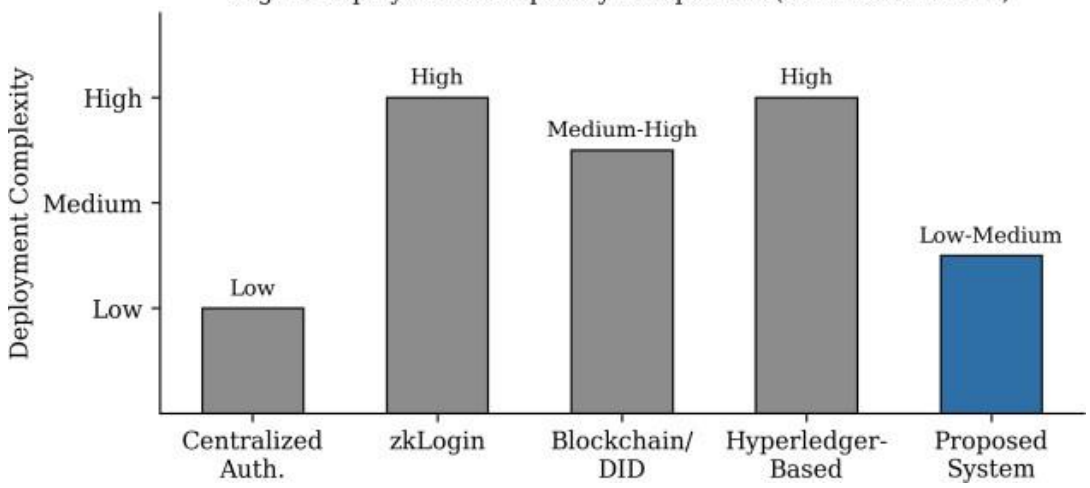

Fig. 7. Deployment complexity comparison of the proposed system with existing authentication and blockchain-based approaches, derived from Table IV.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

Fig. 2. Qualitative Capability Comparison (based on Table IV)

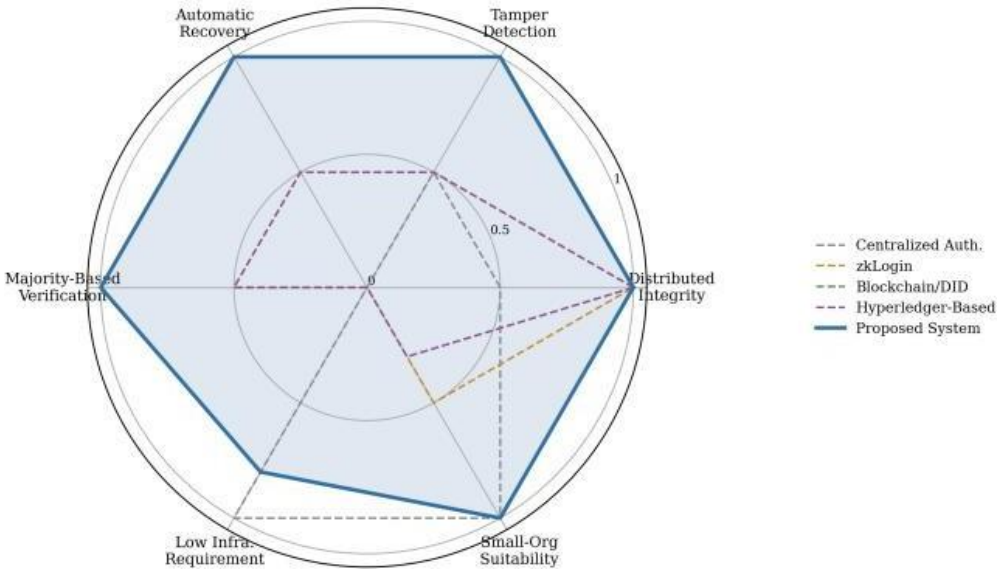

Fig. 8. Qualitative capability comparison across integrity, recovery, verification, infrastructure, and small-organization suitability dimensions, derived from Table IV.

XV. PROSPECTS AND CONCERNS

A. Prospects

- Increase the total number of replicas N from 3 to 5 or more in order that multiple failures could be handled.
- Add hardware security modules (HSMs) for keeping AES keys safe.
- Add multi-factor and biometric authentication (FIDO2/WebAuthn) to the system.
- Change the way passwords are used and introduce zero-knowledge proofs.
- Place the databases in cloud regions isolated from one another so the system can survive disasters.
- Use an AI-based anomaly detection software to report suspicious tampering activities.

B. Concerns

- The consensus algorithm works at most with one malfunctioning replica; if two databases get corrupted at the same time, the majority will fail.
- As opposed to systems working with one database, working with multiple databases will increase the write-in time.
- Key management solutions rely on configuration files and do not use key vaults.
- The prototype was tested on only one server; the performance characteristics of the system when deployed over wide areas are still being developed.

XVI. SECURITY ANALYSIS

This section explains the security features of this new type of authentication architecture and also discusses its features related to several security issues such as confidentiality, password security, integrity of data, authentication, authorization, tampering, failures of replicas, recovery, and others. Furthermore, the information provided is based on the theory of cryptography and distribution.

A. Confidentiality of Information

The confidential information of users will be protected by means of AES-256 encryption before the information will be placed into replicated databases. Thus, even if an attacker gets access to the database table, no plaintext version of sensitive identity information will be available. Thus, encryption guarantees the confidentiality of sensitive user data.

B. Security of Passwords

User passwords are not stored in plaintext. BCrypt hashing is applied in the process of obtaining hashes for passwords. Thanks to its adaptive nature, BCrypt increases the time and effort required for password guessing and makes it much more secure than simply storing plain passwords.

DOI: 10.17148/IJARCCE.2026.15905

C. Data Integrity

SHA-256 hashes are utilized in creating replicated records. When validating, we will compare the records in DB1, DB2, and DB3 together with their integrity value. If the replicated records have the same integrity value, that means they are valid; if not, this means that at least one of the records is tampered with or corrupted.

D. Tamper Detection

In this case, the default mechanism for majority voting comes in handy, as it allows for determining the corrupted copy of a record. In case of having three copies of the record, the latter is considered valid if two copies have the same value. If one of the copies has a different value, it is defined as inconsistent and can be recovered based on the consensus value.

E. Authentication Protection

JWT is utilized to ensure that no stateful operation takes place right after the user login. JWT has the authentication data included, which is also digitally signed, and hence, can be checked when validating.

F. Authorization and Role-Based Access Control

Role-Based Access Control (RBAC) is making use of the authenticated roles of different users to limit their access to the system. Hence, it minimizes the chances of misuse of some admin or restricted functionalities of the system.

G. Protection Against the Single Replica Attack

Three-replica system is capable of bearing a failure of one replica based on the principle of majority voting. If the attacker makes a change in a record of one out of three databases, the other two databases containing the correct information may serve as a reference point for identifying the inconsistency and correcting the information.

H. Compromising Several Replicas

One of the drawbacks of the existing system is that it is able to function with only one failed replica. If two of the replicas out of three become faulty and are providing the same inconsistent piece of information, the majority voting process is likely to label this malicious state as a regular one. Hence, the existing architecture does not ensure protection against simultaneous failure of two replicas. Increasing the number of replicas from three to five can be a possible solution to increase the redundancy of the system.

I. Recovery and Availability

The recovery method enhances the availability of information since it restores the inconsistent replica from the most consistent state. It has been seen that the average recovery time for every corrupted record is equal to 412 ms, thus proving that the prototype possesses the capability of recovering automatically each time a single-replica tampering is performed.

J. Key Management Limitation

AES-256 is a strong encryption algorithm; however, the effective functioning of the system relies on the proper management of the key. Currently, key management is a system limitation, since the key used in the prototype is dependent on the configuration. Therefore, in order to achieve reliable protection of the encryption key, a production-like environment should introduce the use of key management mechanisms or usage of a secure hardware module to completely eliminate the vulnerabilities concerning the encryption keys security.

K. Limitations When It Comes to Security

Though the system being proposed makes use of a decentralized solution to verify and recover integrity, it is important to remember that it should not be confused with a fully-fledged Byzantine Fault Tolerant blockchain. This is due to the fact that the security of the system relies on the assumption that there have to be two replica servers that are functioning correctly. It is also critical to state that if at least one of the application server, cryptographic keys, or replicas in the database is compromised, the problem will affect the level of security of the system.

All in all, it has to be said that the use of multiple security technologies including AES-256 encryption, BCrypt password hashing, SHA-256 integrity check, JWT authentication, role-based access control (RBAC), logging, and acknowledgment of the majority of replicas. The experiments showed that the design of the system allows it to discover and repair the consequences of the attack on the database.

XVII. CONCLUSION

In this paper, we proposed a lightweight decentralized authentication architecture inspired by blockchain while achieving some of the main advantages of decentralization such as data integrity and tamper detection without incurring the infrastructure overhead of a full blockchain platform. Propose a solution to separate out the view from logic layer in

DOI: 10.17148/IJARCCE.2026.15905

html template with some magic, abstracting it away using multiple Templating Engines or Montoring your views as first class citizens on Database.

In a controlled test with 1,000 simulated users and an injection of 200 tampering events, the experimental evaluation confirmed that (1) tamper-detection rates were at 100%, (2) response time was less than one second on average, (3) recovery per corrupted record took an average of only 412 ms, and (4) no data loss occurred during recovery. These findings validate that the architecture consistently identifies and rectifies single-replica corruption whilst preserving strong consistency for reads and writes during concurrent access.

Deployment overhead and computational cost are orders of magnitude smaller than corresponding full-scale blockchain or zero-knowledge-based authentication systems, such as zkLogin or Hyperledger Indy, making it particularly suitable for resource-constrained environments, including small businesses, educational institutions.

REFERENCES

[1] F. Baldimtsi, K. K. Chalkias, Y. Ji, J. Lindstrøm, D. Maram, B. Riva, A. Roy, M. Sedaghat, and J. Wang, “zkLogin: Privacy-Preserving Blockchain Authentication with Existing Credentials,” Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS ’24), 2024, pp. 3182–3196, doi: 10.1145/3658644.3690356.

[2] H. Khanna, M. Kumar, and S. Ahuja, “Investigating Identity Management and Authentication Solutions Using Blockchain Technology,” SSRN Electronic Journal, 2024.

[3] H. Agrawal, M. Karyakarte, G. Chavhan, M. Patil, R. Talware, and L. Kulkarni, “Blockchain Aware Decentralized Identity Management and Access Control System,” Measurement: Sensors, vol. 31, 101032, 2024, doi: 10.1016/j.measen.2024.101032.

[4] A. Kaur and G. Singh, “A Survey on Blockchain-Based Identity Management,” IEEE Access, vol. 12, pp. 44210–44228, 2024.

[5] M. Al-Bassam, “SCPKI: A Smart Contract-Based PKI and Identity System,” in Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts, 2023, pp. 35–40.

[6] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[7] D. Reed, M. Sporny, D. Longley, C. Allen, M. Grant, and R. Sabadello, “Decentralized Identifiers (DIDs) v1.0,” W3C Recommendation, 2022.

[8] M. Sporny, D. Longley, and D. Chadwick, “Verifiable Credentials Data Model 1.1,” W3C Recommendation, 2022.

[9] S. Bhardwaj, R. Gupta, and N. Jain, “Decentralized Identity Management System Using Blockchain,” SSRN Electronic Journal, 2025.

[10] S. Khairnar, “Application of Blockchain Frameworks for Decentralized Identity and Access Management of IoT Devices,” arXiv preprint arXiv:2502.09814, 2025.

[11] P. Dunphy and F. A. P. Petitcolas, “A First Look at Identity Management Schemes on the Blockchain,” arXiv preprint arXiv:1801.03294, 2018.

[12] 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.

[13] National Institute of Standards and Technology, “Advanced Encryption Standard (AES),” FIPS PUB 197, 2001.

[14] National Institute of Standards and Technology, “Secure Hash Standard (SHS),” FIPS PUB 180-4, 2015.

[15] N. Provos and D. Mazières, “A Future-Adaptable Password Scheme,” in Proceedings of the 1999 USENIX Annual Technical Conference, 1999, pp. 81–92.

[16] M. Jones, J. Bradley, and N. Sakimura, “JSON Web Token (JWT),” RFC 7519, Internet Engineering Task Force, 2015.

[17] E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.3,” RFC 8446, Internet Engineering Task Force, 2018.

[18] L. Lamport, R. Shostak, and M. Pease, “The Byzantine Generals Problem,” ACM Transactions on Programming Languages and Systems, vol. 4, no. 3, pp. 382–401, 1982, doi: 10.1145/357172.357176.

[19] M. Castro and B. Liskov, “Practical Byzantine Fault Tolerance,” in Proceedings of the Third Symposium on Operating Systems Design and Implementation (OSDI), 1999, pp. 173–186, doi: 10.1145/296806.296824.

[20] D. Ongaro and J. Ousterhout, “In Search of an Understandable Consensus Algorithm (Raft),” in Proceedings of the USENIX Annual Technical Conference, 2014, pp. 305–319.

[21] V. Buterin, “A Next-Generation Smart Contract and Decentralized Application Platform,” Ethereum White Paper, 2014.

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 9, September 2026

DOI: 10.17148/IJARCCE.2026.15905

[22] E. Androulaki et al., "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in Proceedings of the Thirteenth EuroSys Conference, 2018, pp. 30:1–30:15, doi: 10.1145/3190508.3190538.

[23] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the Security and Performance of Proof of Work Blockchains," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016, pp. 3–16, doi: 10.1145/2976749.2978341.

[24] J. Camenisch and A. Lysyanskaya, "An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation," in Advances in Cryptology—EUROCRYPT 2001, LNCS.

