International Journal of Computer Techniques – Volume 12 Issue 3, May - June - 2025

RESEARCH ARTICLE OPEN ACCESS

BLOCKCHAIN-BASED SECURITY SYSTEM FOR SECURE IDENTITY MANAGEMENT AND ACCESS CONTROL

Vishwas Bhagwat*, Leena Patil**

*(Computer Science and Engineering, Priyadarshini College of Engineering, Nagpur
Email: vishwasamr@gmail.com)

** (Associate Professor, Computer Science and Engineering, Priydarshini College of Engineering, Nagpur
Email: leena.patil@pcenagpur.edu.in)

----------**********----------

Abstract:

This paper presents a blockchain-based security system designed for secure identity management and access control, integrating smart contracts and decentralized file storage. Traditional identity systems rely heavily on centralized architectures, which are prone to data breaches, tampering, and unauthorized access [1], [2]. To address these limitations, the proposed system leverages Ethereum smart contracts to enforce role-based access control and utilizes the InterPlanetary File System (IPFS) to store files with verifiable integrity through content addressing [3], [4]. A Flask-based backend interfaces with both blockchain and IPFS layers, while a responsive frontend built with HTML and Bootstrap provides intuitive interaction for users and administrators. Unlike conventional blockchain solutions that require browser extensions or wallet setup, this system internally manages wallet creation and blockchain transactions, enabling seamless registration, authorization, and file submission. Users are assigned roles that determine access privileges, and all actions—such as file uploads and role changes—are immutably logged on the blockchain. The system is fully deployable on local infrastructure using Ganache and IPFS Desktop, making it ideal for academic environments, secure intranet applications, and proof-of-concept deployments. The results demonstrate that blockchain and IPFS can be combined effectively to deliver a secure, auditable, and decentralized identity management system with minimal deployment complexity.

Keywords — **Blockchain, Identity Management, Access Control, Smart Contracts, IPFS, Decentralized Storage, Flask.**

----------**********----------

I. INTRODUCTION

Digital systems today are increasingly reliant on robust Identity and Access Management (IAM) frameworks to secure data, regulate permissions, and establish trust. Traditionally, these IAM systems operate in centralized environments where identity data, authentication logic, and authorization policies are managed by a central authority. While widely adopted, centralized IAM architectures expose organizations to significant risks such as data breaches, manipulation by insiders, and loss of data integrity due to single points of failure [1].

Recent security incidents have illustrated the shortcomings of such centralized models. The LinkedIn breach in 2021 compromised over 700 million user records, and the 2022 Okta authentication service breach demonstrated that even security-focused platforms can be exploited [2]. These events underscore the urgent need to explore decentralized alternatives that eliminate central points of trust and ensure tamper-evident, verifiable access control mechanisms.

Blockchain technology introduces a decentralized and immutable ledger system, capable of recording transactions and enforcing rules through smart contracts—self-executing scripts stored on the blockchain [3], [5]. These contracts allow for role-based logic enforcement, audit trails, and cryptographic assurance without reliance on intermediaries. When combined with the InterPlanetary File System (IPFS)—a distributed, content-addressed storage layer—it becomes possible to design systems that are not only decentralized in logic but also in data storage [4].

This paper proposes a complete, locally deployable security system that integrates Ethereum-based smart contracts with IPFS to provide decentralized identity management and access control. The system employs Flask as the backend middleware to manage interactions with blockchain and storage components, while a user-friendly web interface enables real-time file uploads, user role assignments, and authorization workflows. Unlike many blockchain solutions that depend on browser wallet plugins, this system autonomously handles wallet generation and management to reduce onboarding complexity [6].

By decentralizing both control and storage layers, the system achieves strong guarantees of data integrity, user auditability, and operational transparency. Designed for academic, institutional, and small enterprise use, this implementation illustrates the real-world feasibility of blockchain-IPFS hybrid architectures for secure identity management.

II. RELATED WORK

Conventional Identity and Access Management (IAM) systems rely on centralized architectures, where a single authority manages user authentication, authorization, and role allocation. While widely adopted, these centralized systems face inherent limitations such as vulnerability to insider threats, single points of failure, and lack of audit transparency [1], [7]. Incidents like the Equifax breach and repeated credential leaks from centralized databases demonstrate the risks posed by relying on such trust models.

Blockchain technology has emerged as a promising alternative for decentralized IAM, offering an immutable and distributed ledger for logging access events and identity credentials. Platforms like Ethereum enable the deployment of smart contracts—automated scripts that enforce policies and roles without intermediary intervention [3]. Several studies have explored the use of blockchain to facilitate role-based access control, verifiable credential issuance, and audit trails. For example, Zyskind et al. proposed a blockchain framework for user-centric identity control [8], while Al-Bassam introduced a smart contract-based public key infrastructure (SCPKI) that eliminates reliance on traditional certificate authorities [9].

However, blockchain alone is insufficient for storing large data payloads such as identity documents. To address this, decentralized file systems like IPFS are often used in tandem. IPFS stores files as cryptographic hashes (CIDs), ensuring content-addressed retrieval and tamper-evident integrity [4]. Researchers have implemented hybrid models where IPFS stores documents off-chain and blockchain stores CIDs, creating a link between identity metadata and verifiable content [10].

Despite these advancements, significant challenges remain. Many proposed systems assume users can manage cryptographic keys or interact with low-level blockchain tools like MetaMask. Additionally, integration between IPFS and blockchain is often shallow—limited to CID logging without access control. Few systems provide full-stack prototypes with frontend, backend, and smart contract layers integrated into a cohesive IAM solution [11].

This work builds on the foundation of prior research by addressing these usability and architectural gaps. It introduces a fully functional IAM system that automates wallet management, integrates IPFS-based storage, and provides a web-based interface with role-based access, administrative controls, and transaction traceability. Unlike earlier prototypes, this system is designed for localized, offline deployment, eliminating the need for public networks or browser extensions

while maintaining the core benefits of decentralization.

III. PROPOSED SYSTEM ARCHITECTURE

The proposed system consists of four core components: the smart contract layer, decentralized storage layer, backend middleware, and the frontend interface. These components collectively enable decentralized identity registration, role-based access control, and secure file storage and retrieval [4], [6].

A. Smart Contract Layer

Smart contracts written in Solidity define the logic for user roles, file registration, and administrative control. Each user is assigned a role (e.g., pending, approved, admin), and functions are guarded using access modifiers to restrict actions like approving users or uploading files to specific roles [5]. The contract also emits events on each action, enabling real-time tracking of system activity. It is deployed on a local Ethereum blockchain using Ganache, which simulates a full-featured blockchain environment with no transaction costs [11].

B. Decentralized Storage Layer

The system integrates IPFS to store files in a tamper-evident and content-addressable manner. Files uploaded by users are hashed and stored in IPFS, generating a Content Identifier (CID). This CID is recorded on the blockchain, establishing a permanent and verifiable link between the user and the uploaded content [4]. IPFS nodes are run locally to ensure full control over data availability during testing [10].

C. Middleware Layer

A Flask backend serves as the middleware between the frontend and the decentralized layers. It handles user registration, login sessions, IPFS uploads, and smart contract interactions. It uses Web3.py for blockchain communication and ipfshttpclient for IPFS integration [6]. The backend also performs error handling and session validation, ensuring only authorized users can access specific features.

D. Frontend Interface

The user interface is built using HTML, Bootstrap, and JavaScript, providing responsive and role-aware navigation. Users can register, log in, upload files, and view transaction history. Admins can approve or promote users and monitor system activity. The interface displays real-time status messages, CID results, and feedback from both IPFS and blockchain events.

This layered architecture ensures modularity, maintainability, and a clear separation of concerns, enabling users and administrators to interact securely and intuitively with the system [11].

IV. IMPLEMENTATION AND RESULTS

The implementation phase translated the architectural design into a functioning prototype deployed locally using Ganache and IPFS Desktop [11]. The smart contract was developed and tested in Remix IDE and then deployed using Web3.py. Flask served as the middleware, exposing REST endpoints for user registration, file upload, and role-based actions. The frontend was designed with dynamic rendering to adapt based on the user's role.

Key functionalities implemented include:

- Automatic wallet assignment from Ganache during registration [6]
- Admin dashboard for approving, promoting, and revoking users
- IPFS-based file storage with CID generation and logging [4], [10]
- Role-restricted access to contract functions
- Flash messages and transaction hash displays on the frontend

During testing, each user operation resulted in a corresponding blockchain transaction. Uploading a file triggered CID generation through IPFS and stored that identifier immutably via a smart contract call [4]. Admin users could verify these transactions

in Ganache's block explorer, and users could retrieve files using their CIDs.

Error scenarios such as unauthorized uploads, invalid CID entries, and inactive IPFS daemon responses were also handled gracefully, providing informative messages and preserving system integrity. The system demonstrated low latency, seamless workflow, and verifiability of all access and storage actions.

These results affirm the viability of combining blockchain and IPFS for identity and access control in a self-contained, deployable system without reliance on public blockchain infrastructure or browser-based wallets [3], [6].

V. DISCUSSION

The developed prototype demonstrates the practical feasibility of using blockchain and IPFS to construct a decentralized identity management system [3], [4]. Through smart contracts, the system successfully enforces role-based access control while maintaining tamper-evident logs of user activity [5], [8]. IPFS complements this by providing verifiable, content-addressed file storage, thereby eliminating reliance on centralized file servers.

Compared to traditional IAM solutions, this system introduces architectural decentralization, which mitigates common vulnerabilities such as data manipulation and single points of failure [1], [7]. Additionally, it differs from most existing blockchain-based IAM implementations by abstracting complex operations such as wallet creation and transaction signing, offering a simplified interface for non-technical users [6].

A critical insight is that decentralization does not necessarily require technical complexity for the end user. By handling blockchain and IPFS logic within the backend, the system achieves usability without compromising decentralization principles. Admins interact through a dedicated dashboard, while general users experience a smooth file upload and verification process with real-time feedback.

However, limitations exist. The use of Ganache restricts real-world deployment due to its ephemeral state [11]. Passwords are stored in plaintext, and IPFS integration is limited to local nodes [10]. The absence of encryption for uploaded files may also limit use cases requiring data confidentiality. Nonetheless, these issues are largely implementation choices rather than architectural flaws, and future enhancements could easily incorporate password hashing, persistent blockchains, and encryption modules.

The results show promise for academic, small-scale enterprise, and institutional use. They also provide a foundational base for more advanced decentralized IAM solutions that might include biometric identity verification, self-sovereign identity (SSI) standards, or integration with public blockchain testnets [8], [9].

VI. CONCLUSION AND FUTURE WORK

This paper presented a blockchain-based security system for decentralized identity management and access control. By integrating Ethereum smart contracts with IPFS and encapsulating backend logic in a Flask-powered middleware, the system ensures tamper-evident, decentralized, and role-restricted access to identity data and file storage. Unlike existing solutions, it removes reliance on external wallets and browser extensions by internally handling transaction logic and wallet creation.

The system was successfully deployed on a local blockchain using Ganache and tested with real user flows involving file uploads, role changes, and CID verifications. Results demonstrated both the technical viability and usability of the platform, supporting its application in controlled environments such as academic institutions, secure intranets, or low-trust collaborative systems.

Future improvements could involve deploying the system on a persistent testnet like Goerli, encrypting files before upload, and integrating biometric authentication or decentralized identifiers (DIDs) aligned with self-sovereign identity frameworks. Additional enhancements to the frontend and automated admin notifications could further improve usability and system responsiveness. The work lays a strong foundation for fully decentralized and user-friendly identity management solutions suitable for real-world deployments.

REFERENCES

[1] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain Technology: Beyond Bitcoin," Applied Innovation Review, no. 2, pp. 6–10, 2016.

[2] D. Goodin, "LinkedIn breach reportedly exposes data of 700 million users," Ars Technica, Jun. 2021.

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

[4] J. Benet, "IPFS - Content Addressed, Versioned, P2P File System," IPFS White Paper, 2021.

[5] N. Atzei, M. Bartoletti, and T. Cimoli, "A Survey of Attacks on Ethereum Smart Contracts (SoK)," in Proc. Int. Conf. Principles of Security and Trust, Springer, pp. 164–186, 2017.

[6] M. Zhang, J. Ma, and W. Chen, "Blockchain and IPFS-Based System for Secure Medical Record Management," IEEE Trans. Emerging Topics in Computing, early access, 2022.

[7] A. Rejeb, K. Rejeb, S. J. Simske, and zH. Treiblmaier, "Blockchain Technology in Supply Chain Transparency: A Literature Review," Technol. Forecast. Soc. Change, vol. 173, 2021.

[8] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing Privacy: Using Blockchain to Protect Personal Data," in Proc. IEEE S&P Workshops, pp. 180–184, 2015.

[9] M. Al-Bassam, "SCPKI: A Smart-Contract-Based Public Key Infrastructure," arXiv preprint arXiv:2104.04242, 2021.

[10] Protocol Labs, "InterPlanetary File System (IPFS)," 2021.

[11] Truffle Suite, "Ganache: Personal Blockchain for Ethereum Development," 2023.