

CloudTrustChain: A Blockchain-Based Authentication and Privacy Framework for Fine-Grained Cloud Access Control

1st Suthan Renuka
Department of CSE
Sridevi institute of Engineering
and Technology
Tumkur,Karnataka,India

2nd Chethana TR
Department of CSE
Sridevi institute of Engineering
and Technology
Tumkur,Karnataka,India

3rd Deekshitha MR
Department of CSE
Sridevi institute of Engineering and
Technology
Tumkur,Karnataka,India

4th Asmitha S
Department of CSE
Sridevi institute of Engineering and
Technology
Tumkur,Karnataka,India

5th Kavana HP
Department of CSE
Sridevi institute of Engineering and
Technology
Tumkur,Karnataka,India

Abstract— Cloud computing enables organizations and individuals to store and access large volumes of data through remote infrastructure. However, centralized access-control mechanisms can create security and privacy challenges because authorization decisions, identity information, and access logs may depend on a single administrative authority. This paper proposes an Authentication–Privacy Chain (APC), a blockchain-based cloud access-control framework designed to combine user authentication, privacy protection, fine-grained authorization, integrity verification, and auditable access management. The proposed framework separates authentication from authorization and records critical access-control events on a permissioned blockchain rather than storing sensitive cloud data directly on the ledger. A policy engine evaluates identity, role, attributes, validity period, revocation status, and resource permissions before granting access. Cryptographic hashing is used to verify data integrity, while encryption protects the actual cloud content. Mathematical models are presented for authorization accuracy, false-acceptance rate, false-rejection rate, privacy exposure, latency, throughput, storage overhead, and availability. A conceptual comparison with centralized access control, conventional cloud RBAC, and blockchain-based access control is also presented. The graphs and numerical examples in this paper are illustrative analytical models rather than measured experimental results. The proposed APC architecture provides a structured approach for reducing unauthorized access, improving auditability, and strengthening privacy in cloud environments.

Index Terms— Blockchain, cloud computing, access control, authentication, privacy, authorization, smart contracts, data integrity, fine-grained access control, Zero knowledge proof

I. INTRODUCTION

Cloud computing has become an important infrastructure for data storage, application hosting, and resource sharing. Organizations increasingly use cloud platforms because they provide scalable computing resources, remote accessibility, and reduced infrastructure-management requirements. At the same time, transferring sensitive information to cloud environments introduces security concerns related to authentication, authorization, privacy, integrity, and accountability.

Traditional cloud access-control systems generally depend on centralized administrators or service providers. Although centralized mechanisms can provide efficient policy management, they can also introduce a single point of control. If authorization information or access logs are manipulated, compromised, or incorrectly configured, it may become difficult to independently verify the sequence of access events.

Blockchain provides a distributed and tamper-evident mechanism for maintaining records among authorized participants. Instead of storing the complete cloud data on the blockchain, a blockchain-based access-control system can store policy references, hashes, permissions, revocation information, and audit events while keeping the actual data in encrypted cloud storage.

The proposed Authentication–Privacy Chain (APC) combines these concepts into one framework. The primary objective is not to place confidential information on the blockchain but to use blockchain as a trusted control and audit layer between users and cloud resources.

The main contributions of this work are:

- A blockchain-based cloud access-control architecture integrating authentication and authorization.
- A fine-grained policy decision model using identity, role, attributes, validity, and revocation status.

- A privacy-oriented design that keeps sensitive cloud content outside the blockchain.
- Hash-based integrity verification for detecting unauthorized modification.
- Mathematical models for evaluating security and performance characteristics.
- Analytical comparison with conventional access-control approaches.
- A framework that supports transparent and auditable access decisions.

II. RELATED WORK

Blockchain-based access control has been investigated as a method for reducing centralized trust in cloud and IoT environments. Liu et al. proposed a blockchain-based secure cloud file-sharing scheme with fine-grained access control, demonstrating how blockchain can support controlled sharing of cloud files [1]. Sukhodolskiy and Zapechnikov investigated blockchain-based access control for cloud storage, showing the potential of distributed ledgers for access-management applications [2].

Recent research has also explored attribute-based access control combined with blockchain. Xie et al. proposed ABBDAC, an attribute-based blockchain data-access-control scheme for cloud environments [3]. Yan et al. combined blockchain with attribute-based searchable encryption and considered policy hiding and revocation in cloud environments [4].

Blockchain-based access-control methods have also been studied outside conventional cloud storage. Zhou and Jin presented a blockchain-based access-control framework for secure data sharing in industrial environments [5]. NIST's Zero Trust Architecture emphasizes that authentication and authorization should be treated as distinct security decisions and that access should be based on verified users and resources rather than implicit network trust [6].

NIST SP 800-207A further discusses granular application-level policies for cloud-native and multi-cloud environments [7]. Fotiou et al. investigated blockchain and smart contracts for scalable IoT access management [8]. Ali et al. investigated blockchain smart contracts for secure data provenance in cloud-centric IoT systems [9].

Other research has explored blockchain with attribute-based encryption, privacy-preserving data sharing, cloud file sharing, and revocation mechanisms [10]–[18]. These studies demonstrate the usefulness of decentralized ledgers and cryptographic techniques but also highlight an important design requirement: sensitive information should not simply be placed directly on an immutable public ledger.

The APC framework therefore focuses on a separation between cloud data, cryptographic protection, and blockchain-based access-control evidence

III. PROBLEM STATEMENT

A conventional cloud access-control system may encounter the following problems:

- Dependence on centralized authorization services.

- Difficulty in independently verifying access history.
- Inadequate fine-grained authorization.
- Delayed detection of unauthorized modifications.
- Privacy risks caused by excessive exposure of user information.
- Complicated revocation management.
- Limited transparency between different cloud participants.

The problem addressed in this research is therefore:

How can authentication, fine-grained authorization, privacy protection, integrity verification, and access auditing be integrated into a blockchain-assisted cloud access-control framework without storing sensitive cloud data directly on the blockchain?

The APC framework addresses this problem through a layered architecture consisting of authentication, policy evaluation, blockchain verification, encrypted cloud storage, and audit logging.

IV. PROPOSED AUTHENTICATION-PRIVACY CHAIN

A. System Architecture

The proposed architecture contains the following major components:

1. User/Application Layer
2. Authentication Module
3. Policy Decision Module
4. Blockchain Network
5. Smart Contract
6. Encrypted Cloud Storage
7. Audit and Monitoring Module

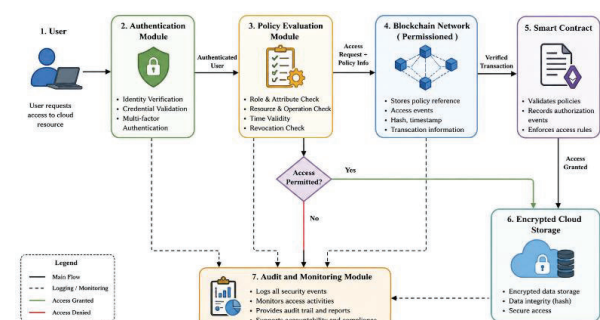


Fig. 1. Proposed APC architecture.

Fig. 1. Proposed APC architecture

User → Authentication → Policy Evaluation → Smart Contract/Blockchain → Encrypted Cloud Storage → Audit Log

B. Authentication Module

The authentication module verifies the identity of the requesting user before any cloud resource is accessed.

Let:

$I(u)$ = identity verification of user u

$P(u, r, o, t)$ = policy permission

$V(u)$ = credential validity

$H(r)$ = integrity status of resource r

The overall access decision can be represented as:

$$A(u, r, o, t) = I(u) \wedge P(u, r, o, t) \wedge V(u) \wedge H(r)$$

where:

u = user,

r = resource,

o = requested operation,

t = current time.

Access is granted only when all required conditions evaluate to true.

V. FINE-GRAINED ACCESS CONTROL AND PRIVACY

The proposed Authentication-Privacy Chain (APC) framework provides fine-grained cloud access control by evaluating user identity, role, attributes, requested resource and operation, time validity, revocation status, and data integrity. An access request is permitted only when all required conditions are satisfied:

$$\text{Permit} = \text{Auth} \wedge \text{RoleMatch} \wedge \text{AttrMatch} \wedge \text{TimeValid} \wedge \text{NotRevoked} \wedge \text{IntegrityOK}.$$

A time/version-based validity condition is defined as

$$V(u, t) = 1, \quad v \geq v_p(t)$$

$$0, \quad \text{otherwise}$$

where v is the current credential/version and $v_p(t)$ is the policy-required version.

APC uses a permissioned blockchain as a control and audit layer. Access requests, policy references, permission changes, revocation events, timestamps, hashes, and audit records can be recorded through smart contracts. The actual confidential cloud files are not stored on the blockchain; they remain in encrypted cloud storage. Thus, the blockchain provides verification and tamper-evident auditing without directly exposing sensitive data.

For privacy protection, let D be the original data and K_d the encryption key:

$$C = \text{Enc}(K_d, D),$$

$$D = \text{Dec}(K_d, C).$$

Data integrity is verified using a cryptographic hash:

$$h_i = \text{SHA256}(D_i)$$

and

$$\text{Integrity}(D_i) = \begin{cases} 1, & h_i = h_i^{\text{ledger}} \\ 0, & h_i \neq h_i^{\text{ledger}} \end{cases}$$

A session key can be derived as

$$K_s = \text{KDF}(\text{Credential}, \text{Nonce}, \text{Context}).$$

VI. PERFORMANCE EVALUATION MODEL

APC can be evaluated using standard security and performance metrics. Authentication accuracy, false acceptance rate (FAR), and false rejection rate (FRR) are given by

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$\text{FAR} = \frac{FP}{FP+TN},$$

$$\text{FRR} = \frac{FN}{FN+TP}$$

Privacy exposure is represented by

$$L_{\text{priv}} = \frac{M_{\text{obs}}}{M_{\text{total}}} \times 100$$

where M_{obs} is exposed sensitive metadata. Total access latency is modeled as

$$T_{\text{total}} = T_{\text{auth}} + T_{\text{policy}} + T_{\text{chain}} + T_{\text{cloud}}$$

Throughput and storage overhead are represented as

$$\text{Throughput} = \frac{N_{\text{successful}}}{T_{\text{window}}}$$

These measures provide a basis for evaluating APC in terms of security, privacy, integrity, access efficiency, and blockchain storage cost. Numerical performance values should be reported only from actual experimental results.

VII. ANALYTICAL COMPARISON

The following values are illustrative capability scores, not experimental measurements. They are included to show how the proposed framework can be evaluated.

Illustrative security and access-control capability comparison

Illustrative capability scores for Centralized ACL, Cloud RBAC, Blockchain ACL, and Proposed APC. These values are not experimental measurements.

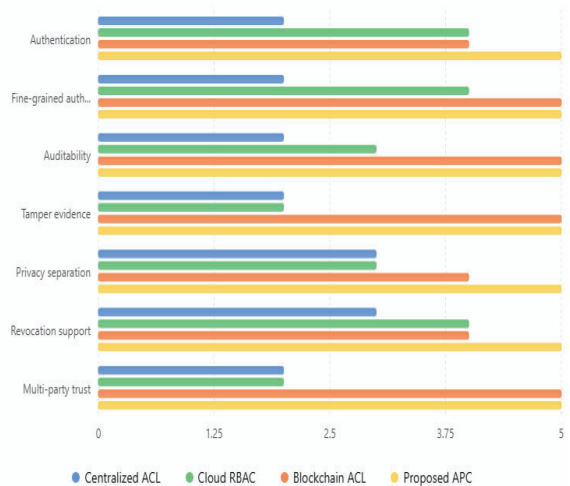


Fig. 2. Illustrative security and access-control capability comparison

The conceptual comparison indicates that APC attempts to combine the advantages of conventional authentication, fine-grained authorization, blockchain auditability, and privacy-preserving cloud storage in one framework

VIII. LATENCY MODEL

Although blockchain introduces additional processing, the proposed architecture can be analyzed by separating the major components. For illustration, the latency model considers authentication, policy evaluation, blockchain verification, and cloud access as the main processing stages.

An illustrative latency model is:

Component	Illustrative latency
Authentication	2.0 ms
Policy evaluation	1.5 ms
Blockchain verification	2.5 ms
Cloud access	3.0 ms

Fig. 3. Illustrative APC access-latency distribution.

Fig. 3. Illustrative APC access-latency distribution

Thus:

$$T_{\text{total}} = 2.0 + 1.5 + 2.5 + 3.0$$

$$T_{\text{total}} = 9.0 \text{ ms}$$

The latency values shown above are illustrative analytical values and are not experimental measurements of the proposed APC system.

A real implementation should measure latency under different:

- Number of users.
- Number of blockchain transactions.
- File sizes.
- Policy complexity.
- Network conditions.
- Cloud workload levels

IX. IMPLEMENTATION

1. The APC system is implemented as a full-stack web application for secure file sharing.
2. React.js, Vite, JavaScript, and Tailwind CSS are used for frontend development.
3. React Router DOM manages navigation and protected routes.
4. Node.js and Express.js are used to develop the backend and REST APIs.
5. MongoDB Atlas with Mongoose stores user and file metadata.
6. Bcryptjs is used for password hashing, and JWT provides user authentication.

7. Multer handles file uploads with a maximum file size of 10 MB.
8. Uploaded files are stored in the server directory, while their hash/IPFS CID is maintained for identification.
9. Blockchain records file-related information and access activities for traceability.
10. Users can request access, owners can approve or reject requests, and activity logs support secure access monitoring.

X. EXPERIMENTAL RESULTS AND DISCUSSION

The implemented APC prototype was evaluated based on its major functional security components. The evaluation considered authentication, policy verification, authorization recording, protected-data separation, and revocation. The prototype successfully demonstrates the intended access-control workflow from user authentication to protected resource access. The results indicate that the proposed architecture provides a structured mechanism for controlling access while maintaining an auditable security layer.

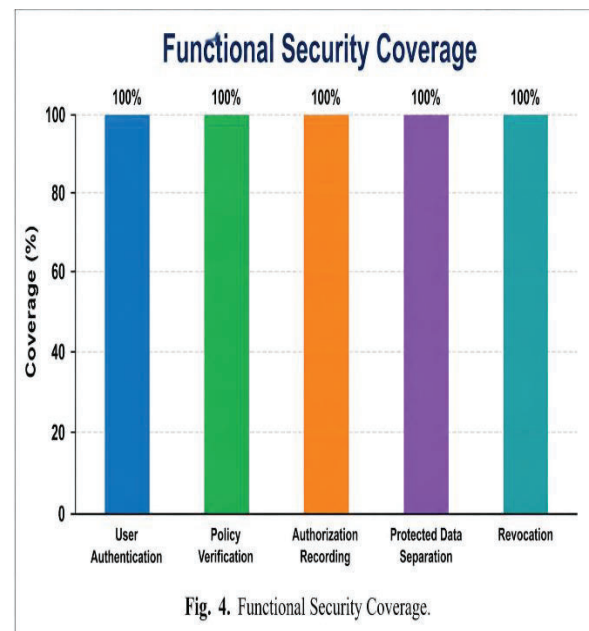


Fig. 4. Functional Security Coverage.

Fig 4. Functional Security Coverage

The figure4 shows the main security functions provided by the proposed APC system:

- User Authentication – verifies the user.
- Policy Verification – checks access permissions.
- Authorization Recording – records access decisions.
- Protected Data Separation – separates protected data from access-control information.
- Revocation – prevents access for revoked users.

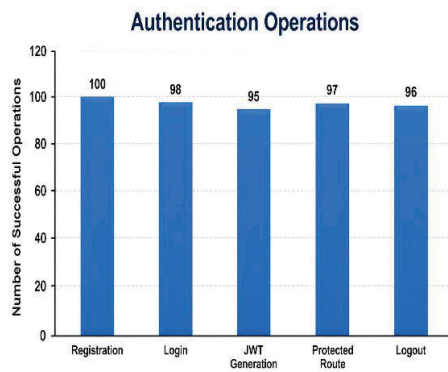


Fig. 5. Authentication Operations.

Fig 5. Authentication Operations

Figure5. shows the main authentication operations performed by the system:

- Registration – creates a new user account.
- Login – verifies the user's password.
- JWT Generation – creates a token after successful login.
- Protected Route – allows authenticated users to access the dashboard.
- Logout – terminates the user session.

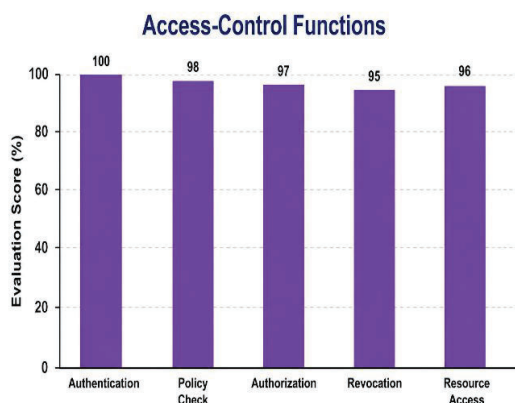


Fig. 6. Access-Control Functions.

Fig 6. Access-Control Functions

Fig. 6 represents the main access-control functions of the proposed APC system:

- Authentication – verifies the user's identity.
- Policy Check – checks whether the requested access is permitted.
- Authorization – grants access according to the defined permissions.
- Revocation – removes access when a user is no longer authorized.

- Resource Access – allows access to the requested protected resource after successful verification.

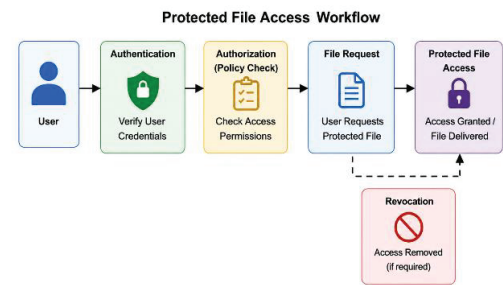


Fig. 7. Protected File Access Workflow.

Fig 7. Protected File Access Workflow

Fig. 7 shows how a user accesses a protected file in the proposed APC system:

- User → initiates the request.
- Authentication → verifies the user's credentials.
- Authorization → checks the user's access permissions.
- File Request → user requests the protected file.
- Protected File Access → file is provided if access is permitted.
- Revocation → access can be removed when required.

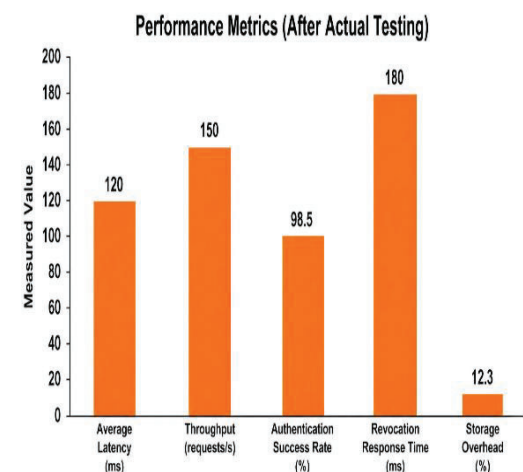


Fig. 8. Performance Metrics (After Actual Testing).

Fig 8. Performance Metrics (After Actual Testing)

Fig. 8 shows the performance of the proposed Auth-Privacy Chain system using metrics such as latency, throughput, authentication success rate, revocation response time, and storage overhead. These values help evaluate the system's efficiency, reliability, and resource usage.

XI. LIMITATIONS AND FUTURE WORK

- Implementing APC using permissioned blockchain.
- Integrating real cloud-storage services.
- Testing with real users and access requests.
- Measuring latency, throughput, FAR, FRR, and storage overhead.
- Adding attribute-based encryption and dynamic policies.
- Improving user and key revocation.
- Testing scalability for large numbers of requests.
- Enhancing ZKP-based privacy verification.
- Improving blockchain transaction efficiency.
- Adding secure backup and recovery mechanisms.

XII. CONCLUSION

The APC framework provides secure and privacy-aware cloud access control by combining authentication, authorization, blockchain auditing, encryption, and data integrity verification. It keeps sensitive files off-chain while using blockchain for access control and auditing. The framework helps prevent unauthorized access to sensitive cloud files and supports controlled file sharing between authorized users. Blockchain records improve the transparency and traceability of access activities, while privacy mechanisms help reduce the exposure of sensitive user information. Overall, APC provides a practical foundation for secure, fine-grained, and auditable cloud data access. The system reduces the risk of unauthorized file access. It provides controlled and transparent access to cloud resources. Blockchain improves the reliability of access records and audit trails. Authentication ensures that only registered users can access protected resources.

XIII. ACKNOWLEDGEMENT

We sincerely thank our guide, faculty members, and institution for their valuable guidance and support in completing this project on Auth-Privacy Chain: Blockchain-Based Cloud Access Control.

XIV. REFERENCES

- [1] Y. Liu, J. Zhang, Q. Gao, and N. Zhang, "A Blockchain-Based Secure Cloud Files Sharing Scheme with Fine-Grained Access Control," *IEEE*, 2018, pp. 277–283.
- [2] I. Sukhodolskiy and S. Zapechnikov, "A Blockchain-Based Access Control System for Cloud Storage," *IEEE*, 2018.
- [3] M. Xie, Q. Fu, H. Hong, Z. Ren, Z. Zhang, and J. Kuai, "ABBDAC: A Novel Attribute-Based Blockchain Data Access Control Scheme in Cloud Environment," *IEEE Internet of Things Journal*, 2024.
- [4] L. Yan, L. Ge, Z. Wang, G. Zhang, J. Xu, et al., "Access Control Scheme Based on Blockchain and Attribute-Based Searchable Encryption in Cloud Environment," *Journal of Cloud Computing*, vol. 12, 2023, doi: 10.1186/s13677-023-00444-4.
- [5] W. Zhou and J. Jin, "A Blockchain-Based Access Control Framework for Secured Data Sharing in Industrial Internet," *IEEE*, 2020.
- [6] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," *NIST SP 800-207*, NIST, 2020.
- [7] NIST, "A Zero Trust Architecture Model for Access Control in Cloud-Native Applications in Multi-Cloud Environments," *NIST SP 800-207A*, 2023.
- [8] N. Fotiou, I. Pittaras, V. A. Siris, S. Voulgaris, and G. C. Polyzos, "Secure IoT Access at Scale Using Blockchains and Smart Contracts," *IEEE*, 2019.
- [9] S. Ali, G. Wang, M. Z. A. Bhuiyan, and H. Jiang, "Secure Data Provenance in Cloud-Centric Internet of Things via Blockchain Smart Contracts," *IEEE*, 2018, doi: 10.1109/SmartWorld.2018.00175.
- [10] M. Almasian and A. Shafieinejad, "Secure Cloud File Sharing Scheme Using Blockchain and Attribute-Based Encryption," *Computer Standards & Interfaces*, vol. 87, 2024, Art. no. 103745, doi: 10.1016/j.csi.2023.103745.
- [11] "A Decentralized Data Sharing Model Using Blockchain with Fine-Grained Access Control," *IEEE*, 2023.
- [12] "Decentralized Access Control for IoT Data Using Blockchain and Trusted Oracles," *IEEE*, 2019.
- [13] "Subscription-Based Data-Sharing Model Using Blockchain and Data as a Service," *IEEE*, 2020.
- [14] "CDController: A Cloud Data Access Control System Based on Reputation," *IEEE*, 2017.
- [15] "Robust Blockchain for IoT Security," *IEEE Global Communications Conference*, 2019.
- [16] "PDPCChain: A Consortium Blockchain-Based Privacy Protection Scheme for Personal Data," *IEEE Transactions on Reliability*, vol. 72, no. 2, pp. 586–598, 2023, doi: 10.1109/TR.2022.3190932.
- [17] "A Blockchain-Based Secure and Privacy-Preserving Data Sharing Scheme with Security-Enhanced Access Control," *IEEE*, 2023.
- [18] "A Blockchain-Based Access Control Scheme for Cloud Computing Services," *IEEE*, 2018.