COMPUTER SCIENCE BULLETIN

VOL: 09 ISSUE: 01 (2026)

E-ISSN: 2959-5347 P-ISSN: 2959-5339

Blockchain-Based Authentication and Access Control for Secure Cloud Computing Infrastructure

Adrian Weller

Department of Engineering, University of Cambridge, Cambridge, UK

Abstract. *Cloud computing concentrates valuable workloads behind identity services, administrative interfaces, and machine-to-machine authorization channels. Centralized access records can be difficult to reconcile across providers and may create single points of administrative failure. Blockchain-based designs offer tamper-evident policy history, shared verification, and programmable authorization, but they also introduce consensus delays, governance burdens, privacy exposure, and smart-contract risks. This narrative review evaluates an architecture in which permissioned ledger services register identity commitments and access-policy versions while sensitive attributes, credentials, and workload data remain off-chain. The paper discusses decentralized identifiers, verifiable credentials, challenge-response authentication, role- and attribute-based authorization, revocation, privacy-preserving evidence, cloud integration, and realistic security evaluation. Rather than presenting invented performance measurements, it proposes testable research questions, threat scenarios, and comparison baselines for assessing whether a ledger materially improves security over conventional identity and access management. The principal conclusion is that blockchain can improve cross-organizational auditability in some settings but is not a substitute for secure identity proofing, key custody, sound policy engineering, or operational monitoring.*

Keywords: *Blockchain Security, Cloud Computing, Authentication, Access Control, Decentralized Identity*

Introduction

Cloud computing has emerged as a fundamental component of modern digital infrastructure, providing scalable, flexible, and cost-effective computing resources to individuals, enterprises, and government organizations. The increasing adoption of cloud-based services has transformed data storage, application deployment, resource management, and information sharing across distributed environments. However, the rapid expansion of cloud computing has also introduced significant security challenges, particularly concerning user authentication, access control, data confidentiality, identity management, and unauthorized resource utilization. Traditional centralized authentication and authorization mechanisms often depend on trusted third-party authorities, creating potential vulnerabilities such as single points of failure, credential

compromise, insider threats, and unauthorized access. These limitations highlight the growing need for decentralized, transparent, and tamper-resistant security frameworks capable of protecting sensitive cloud resources.Blockchain technology offers a promising approach to addressing these challenges through its decentralized architecture, cryptographic security mechanisms, distributed consensus protocols, and immutable transaction records. Unlike conventional centralized security systems, blockchain enables authentication and access control operations to be recorded and verified across distributed network participants, reducing dependence on a single controlling authority. Smart contracts further enhance blockchain-based security by automating authorization policies, verifying access permissions, and enforcing predefined security rules without requiring continuous manual intervention. The integration of blockchain with cloud computing can strengthen identity verification, improve auditability, and support transparent management of access privileges across multiple cloud service providers.Authentication and access control are essential components of cloud security because they determine whether users, applications, and connected devices are authorized to access specific computing resources. Conventional approaches, including password-based authentication, role-based access control, and centralized identity management, may become increasingly difficult to manage in dynamic multi-cloud and hybrid-cloud environments. Blockchain-based authentication frameworks can incorporate cryptographic identities, digital signatures, decentralized identifiers, and verifiable credentials to establish more secure identity verification mechanisms. Similarly, blockchain-enabled access control systems can employ smart contracts to implement role-based, attribute-based, and policy-driven authorization models while maintaining verifiable records of access requests and decisions.Despite these potential advantages, integrating blockchain into cloud security infrastructure introduces several technical and operational challenges. Blockchain networks may experience scalability limitations, transaction processing delays, computational overhead, interoperability difficulties, and privacy concerns. Public blockchain platforms may expose transaction metadata, while permissioned blockchain networks require effective governance and participant management. Furthermore, vulnerabilities in smart contract implementation and weaknesses in cryptographic key management can compromise security if appropriate safeguards are not established. Therefore, blockchain-based authentication systems must be carefully designed to balance security, performance, usability, and operational efficiency.

Cloud Security and the Identity Trust Problem:

The issue of shared responsibility boundaries matters for cloud security and the identity trust problem because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Identity federation across tenants:

The issue of identity federation across tenants matters for cloud security and the identity trust problem because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original

enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Credential theft and session compromise:

The issue of credential theft and session compromise matters for cloud security and the identity trust problem because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Administrative privilege and insider threats:

The issue of administrative privilege and insider threats matters for cloud security and the identity trust problem because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Audit fragmentation in hybrid clouds:

The issue of audit fragmentation in hybrid clouds matters for cloud security and the identity trust problem because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Security objectives and adversary assumptions:

The issue of security objectives and adversary assumptions matters for cloud security and the identity trust problem because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Blockchain Foundations for Identity Management:

This section assesses the relevant security property, implementation design, adversarial assumptions, and practical evaluation requirements.

Distributed ledgers and replicated state:

The issue of distributed ledgers and replicated state matters for blockchain foundations for identity management because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently

establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Permissioned versus permissionless networks:

The issue of permissioned versus permissionless networks matters for blockchain foundations for identity management because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Consensus and finality:

The issue of consensus and finality matters for blockchain foundations for identity management because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Smart contracts as policy enforcement logic:

The issue of smart contracts as policy enforcement logic matters for blockchain foundations for identity management because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Cryptographic commitments and hashes:

The issue of cryptographic commitments and hashes matters for blockchain foundations for identity management because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Governance and ledger lifecycle:

The issue of governance and ledger lifecycle matters for blockchain foundations for identity management because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Decentralized Authentication Architecture:

This section assesses the relevant security property, implementation design, adversarial assumptions, and practical evaluation requirements.

Identity enrollment and proofing:

The issue of identity enrollment and proofing matters for decentralized authentication architecture because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Public keys and decentralized identifiers:

The issue of public keys and decentralized identifiers matters for decentralized authentication architecture because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Challenge-response authentication:

The issue of challenge-response authentication matters for decentralized authentication architecture because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Verifiable credentials and selective disclosure:

The issue of verifiable credentials and selective disclosure matters for decentralized authentication architecture because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Key recovery and rotation:

The issue of key recovery and rotation matters for decentralized authentication architecture because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Replay resistance and session binding:
The issue of replay resistance and session binding matters for decentralized authentication architecture because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Access Control Models and Policy Design:
This section assesses the relevant security property, implementation design, adversarial assumptions, and practical evaluation requirements.

Role-based access control:
The issue of role-based access control matters for access control models and policy design because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Attribute-based access control:
The issue of attribute-based access control matters for access control models and policy design because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Capability tokens and delegation:
The issue of capability tokens and delegation matters for access control models and policy design because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Least privilege and separation of duties:
The issue of least privilege and separation of duties matters for access control models and policy design because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Context-sensitive access decisions:
The issue of context-sensitive access decisions matters for access control models and policy design because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
Revocation and emergency access:
The issue of revocation and emergency access matters for access control models and policy design because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
Smart Contracts and Authorization Workflows:
This section assesses the relevant security property, implementation design, adversarial assumptions, and practical evaluation requirements.
Policy registration and versioning:
The issue of policy registration and versioning matters for smart contracts and authorization workflows because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
Request verification and authorization:
The issue of request verification and authorization matters for smart contracts and authorization workflows because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
On-chain events and audit evidence:
The issue of on-chain events and audit evidence matters for smart contracts and authorization workflows because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Contract testing and formal reasoning:
The issue of contract testing and formal reasoning matters for smart contracts and authorization workflows because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
Upgradeability and governance controls:
The issue of upgradeability and governance controls matters for smart contracts and authorization workflows because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
Gas costs and transaction contention:
The issue of gas costs and transaction contention matters for smart contracts and authorization workflows because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
Privacy and Cryptographic Protection:
This section assesses the relevant security property, implementation design, adversarial assumptions, and practical evaluation requirements.
Off-chain storage of personal data:
The issue of off-chain storage of personal data matters for privacy and cryptographic protection because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.
Encryption and key custody:
The issue of encryption and key custody matters for privacy and cryptographic protection because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Zero-knowledge proofs and privacy tradeoffs:
The issue of zero-knowledge proofs and privacy tradeoffs matters for privacy and cryptographic protection because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Metadata inference and linkability:
The issue of metadata inference and linkability matters for privacy and cryptographic protection because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Consent and purpose limitation:
The issue of consent and purpose limitation matters for privacy and cryptographic protection because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Retention and deletion conflicts:
The issue of retention and deletion conflicts matters for privacy and cryptographic protection because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Integration with Cloud Infrastructure:
This section assesses the relevant security property, implementation design, adversarial assumptions, and practical evaluation requirements.

Identity providers and single sign-on:
The issue of identity providers and single sign-on matters for integration with cloud infrastructure because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Api gateways and service meshes:
The issue of API gateways and service meshes matters for integration with cloud infrastructure because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Container and kubernetes authorization:
The issue of container and Kubernetes authorization matters for integration with cloud infrastructure because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Cross-cloud interoperability:
The issue of cross-cloud interoperability matters for integration with cloud infrastructure because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Resilience and outage behavior:
The issue of resilience and outage behavior matters for integration with cloud infrastructure because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Legacy migration and phased rollout:
The issue of legacy migration and phased rollout matters for integration with cloud infrastructure because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Evaluation Methodology and Security Metrics:
This section assesses the relevant security property, implementation design, adversarial assumptions, and practical evaluation requirements.

Threat modeling and attack simulation:

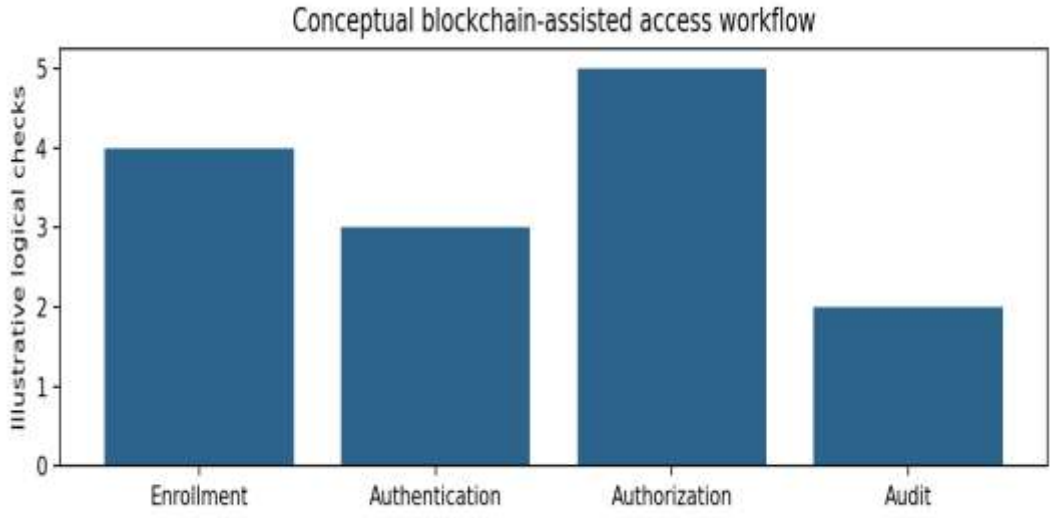
The issue of threat modeling and attack simulation matters for evaluation methodology and security metrics because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Latency and throughput measurement:

The issue of latency and throughput measurement matters for evaluation methodology and security metrics because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.

Availability under node failures:

The issue of availability under node failures matters for evaluation methodology and security metrics because security decisions are only as reliable as their assumptions about identity, authority, and system boundaries. A credible design identifies the principals involved, the protected resource, the action being requested, and the trust relationship between organizations. It must also distinguish what a ledger can prove from what it cannot: an immutable record may show that a credential was registered, but it cannot independently establish that the original enrollment was truthful. Consequently, governance and identity verification remain necessary even when cryptographic validation is automated.



Hypothetical process illustration; not measured latency or performance.

Summary:

Blockchain-based authentication and access control represent a promising direction for enhancing security, transparency, and accountability in modern cloud computing infrastructure. This study examines how decentralized ledger technology, cryptographic identity verification, smart contracts, and distributed authorization mechanisms can address selected limitations of

conventional centralized cloud security systems. Blockchain provides tamper-evident transaction records and verifiable security operations, potentially reducing dependence on centralized authorities while improving the traceability of authentication and access control activities.The integration of decentralized identity management with smart contract-based authorization can support secure access to cloud resources across distributed, hybrid-cloud, and multi-cloud environments. Cryptographic signatures, decentralized identifiers, and verifiable credentials offer mechanisms for strengthening identity verification, while automated authorization policies can improve consistency and accountability in resource access decisions. Additionally, blockchain-supported audit trails can facilitate security monitoring, compliance assessment, and investigation of unauthorized access attempts.However, the effectiveness of blockchain-based cloud security depends on addressing challenges associated with scalability, transaction latency, computational complexity, interoperability, privacy protection, and smart contract vulnerabilities. Blockchain does not inherently eliminate credential theft, compromised endpoints, or poorly configured authorization policies. Effective deployment therefore requires secure key management, appropriate consensus mechanisms, privacy-preserving architectures, and integration with established cloud security controls.

References

- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.
- Androulaki, E., et al. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains. Proceedings of EuroSys.
- Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. Ethereum Yellow Paper.
- Ferraiolo, D. F., & Kuhn, D. R. (1992). Role-based access controls. 15th National Computer Security Conference.
- Sandhu, R., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). Role-based access control models. Computer, 29(2), 38–47.
- Hu, V. C., et al. (2014). Guide to Attribute Based Access Control (ABAC) Definition and Considerations. NIST SP 800-162.
- Hardt, D. (2012). The OAuth 2.0 Authorization Framework. RFC 6749.
- Sakimura, N., Bradley, J., Jones, M., de Medeiros, B., & Mortimore, C. (2014). OpenID Connect Core 1.0.
- W3C. (2022). Decentralized Identifiers (DIDs) v1.0: Core architecture, data model, and representations.
- W3C. (2025). Verifiable Credentials Data Model v2.0. W3C Recommendation.
- OWASP Foundation. (2023). Smart Contract Top 10.
- National Institute of Standards and Technology. (2020). Digital Identity Guidelines: Authentication and Lifecycle Management. NIST SP 800-63B.