Innovative Journal of Computing, Security and Digital Technologies
Volume 1, Issue 2, May – Aug 2026
Page no : 09-20

Blockchain-Enabled Identity Management for Secure Digital Services

Ravi Babu[1], Seshasai Priya Sadam[2]
[1,2] Department of Computer Science & Engineering (AI & ML), G. Narayanamma Institute of Technology and Science, Hyderabad

Abstract

Digital services rely on identity management to authenticate users, authorize access, and maintain trust across organizations. Conventional identity systems often depend on centralized providers that store personal information and control authentication processes. Although centralized and federated approaches are widely deployed, they can create concentrated risks involving data breaches, service outages, excessive disclosure, and limited user control. This paper presents a blockchain-enabled identity management framework designed to support secure access to digital services while limiting the amount of personal information exchanged. The framework combines decentralized identifiers (DIDs), verifiable credentials (VCs), cryptographic proof, smart-contract-based registry functions, and off-chain storage. Trusted issuers provide signed credentials; users hold credentials in digital wallets; and service providers verify presentations without needing to contact the original issuer for every transaction. Blockchain is used to publish identity-related records and credential status information, rather than storing personal identity data directly on-chain. The proposed design includes credential issuance, presentation, verification, revocation, and recovery processes. A threat analysis considers impersonation, credential forgery, replay, unauthorized disclosure, compromised issuers, and ledger-related risks. The paper also defines an evaluation methodology covering authentication latency, verification cost, revocation delay, scalability, and privacy exposure. The framework is intended as a design proposal for digital government, education, healthcare, and financial services. Its practical suitability depends on implementation, interoperability, governance, and empirical evaluation.

Keywords: Blockchain; digital identity; decentralized identifiers; verifiable credentials; authentication; privacy; digital services

Publication Date : 16.05.2026

1. INTRODUCTION

Digital services increasingly require users to prove their identity before accessing information or completing transactions. Examples include applying for government services, accessing academic records, using online banking, and retrieving healthcare information. Identity management therefore serves as a foundational component of digital service delivery. A reliable identity system must establish that a user is who they claim to be, provide appropriate access rights, protect personal information, and support recovery when credentials are lost or compromised.

How to Cite
Blockchain-Enabled Identity Management for Secure Digital Services. (2026). *Innovative Journal of Computing, Security and Digital Technologies*, *1*(2), 09-20.
https://ijcdst.in/index.php/about/article/view/13

Many existing systems use centralized identity providers. In such systems, an organization stores user records and performs authentication on behalf of relying services. Federated identity systems allow a user to authenticate through one provider and access multiple services. These approaches can improve usability and reduce repeated account creation. However, they may also concentrate sensitive data and trust in a limited number of providers. A compromise or outage affecting a central identity provider can disrupt several connected services. Users may also have limited visibility into how their identity attributes are shared and reused.

Blockchain technology has been proposed as a way to support distributed identity management. A blockchain can maintain a shared, tamper-evident record without requiring every participant to operate the same central database. In identity systems, it can support the publication of decentralized identifiers, issuer information, and credential-status records. However, blockchain does not itself establish that a real-world identity is genuine. Trust still depends on the organizations that verify people and issue credentials, as well as the governance rules under which those credentials are accepted.

Decentralized identity management combines blockchain-related infrastructure with cryptographic credentials and user-controlled identity wallets. Decentralized identifiers can identify people or organizations without requiring a single identity provider to control the identifier. Verifiable credentials allow an issuer to make digitally signed claims about a subject. A user can present selected claims to a service provider, which verifies the issuer's signature and applicable status information. Research reviews identify interoperability, self-sovereignty, privacy, and credential lifecycle management as central concerns in decentralized identity systems (Yan et al., 2024; Feng et al., 2025).

This paper proposes a blockchain-enabled identity management framework for secure digital services. The design focuses on five requirements: reliable credential verification, user control over credential presentation, privacy-preserving data handling, support for credential revocation, and interoperability across service providers. The proposed framework does not place personal identity documents or sensitive attributes directly on the blockchain. Instead, the ledger stores only the minimum information needed to support identifier resolution, issuer verification, and credential-status checking.

The main contributions of this paper are:

1. A layered architecture integrating users, credential issuers, service providers, and a permissioned blockchain network.
2. A credential lifecycle covering issuance, storage, presentation, verification, revocation, and recovery.
3. A threat analysis identifying security and privacy risks across the identity lifecycle.
4. An evaluation plan defining measurable criteria for assessing the framework in a prototype implementation.

The remainder of the paper is organized as follows. Section 2 reviews relevant identity management approaches. Section 3 defines the problem and design requirements. Section 4 presents the proposed framework. Section 5 describes the security and privacy analysis.

Section 6 outlines an evaluation methodology. Section 7 discusses limitations and deployment considerations. Section 8 concludes the paper.

2. RELATED WORK

2.1 Centralized and Federated Identity Management

Centralized identity management places identity records and authentication functions under the control of a single organization. It can simplify administration and support consistent access policies. However, it also creates a concentrated point of trust and may expose large amounts of personal information if the identity provider is compromised.

Federated identity management enables multiple service providers to rely on a common identity provider or a trusted group of providers. This model can reduce password fatigue and simplify access across organizational boundaries. Nevertheless, federation still requires trust relationships among participating organizations and may provide limited user control over the attributes released during authentication.

Blockchain-enabled identity management seeks to reduce dependence on a single identity authority by distributing selected identity-management functions. A recent interdisciplinary review describes decentralized identity management in terms of task requirements, stakeholder values, and technical properties such as interoperability and user control (Yan et al., 2024). Decentralization, however, does not remove the need for trusted issuers, clear governance, or secure recovery procedures.

2.2 Decentralized Identifiers and Verifiable Credentials

Decentralized identifiers (DIDs) provide identifiers that can be resolved through defined methods without depending on a single centralized registry. A DID can be associated with a DID document containing public-key and service-endpoint information. The identifier's controller can use the corresponding private key to prove control over the identifier.

Verifiable credentials provide digitally signed claims issued by an organization or authority. A credential may contain attributes such as qualification, membership, or eligibility. A user stores the credential and later presents it to a service provider. The provider checks the signature, issuer, credential structure, validity period, and revocation or status information.

The combination of DIDs and verifiable credentials supports identity systems in which users hold credentials and disclose them when needed. A survey of DID and verifiable credential technologies discusses their use across different application domains and identifies implementation and adoption challenges (Mazzocca et al., 2024). Such systems still require careful handling of private keys, issuer trust, status checking, and attribute disclosure.

2.3 Blockchain-Based Identity and Security Challenges

Blockchain can provide a shared record of public identity information and credential-status updates. A permissioned network may be suitable when participating institutions are known and must follow common governance rules. A public blockchain may offer broader accessibility but can introduce transaction fees, public metadata exposure, and governance complexity.

Prior work has explored blockchain-based identity management and credential verification. For example, research on supervised and revocable decentralized identity proposes

cryptographic methods for privacy protection and credential revocation (He et al., 2024). Other work identifies the need to evaluate interoperability, revocation, scalability, and practical deployment across different application contexts (Feng et al., 2025).

The proposed framework builds on these directions by specifying a service-oriented architecture and a complete identity lifecycle. It treats blockchain as a supporting trust and status layer rather than a repository for personal identity records.

3. PROBLEM DEFINITION AND DESIGN REQUIREMENTS

3.1 Problem Definition

The problem addressed in this paper is how to enable users to authenticate to multiple digital services while reducing dependence on centralized identity storage and limiting unnecessary disclosure of personal information.

The framework considers four principal entities:

- User: the individual or organization seeking access to a digital service.
- Credential issuer: an authorized institution that verifies claims and issues credentials.
- Service provider: a digital service that requests and verifies identity claims.
- Blockchain network: a shared infrastructure that maintains selected public identity and credential-status records.

The framework assumes that issuers are authorized through a governance process and that users can securely operate a digital wallet. It does not assume that blockchain participants are automatically trustworthy or that a valid digital signature guarantees the truth of an issued claim.

3.2 Design Requirements

The framework is designed around the following requirements.

R1 — Authenticity: Service providers must be able to verify that a credential was issued by an authorized issuer and has not been altered.

R2 — User control: Users should hold their credentials and decide when to present them, subject to applicable service and legal requirements.

R3 — Data minimization: A service provider should receive only the attributes required for the requested transaction.

R4 — Revocation support: Issuers must be able to mark credentials as invalid, and service providers must be able to check credential status.

R5 — Interoperability: The framework should support common identity and credential formats so that credentials can be used across participating services.

R6 — Availability and recovery: The system should provide procedures for key loss, device replacement, and compromised credentials.

R7 — Auditability: Authorized parties should be able to inspect relevant system events without exposing unnecessary personal information.

3.3 Scope and Assumptions

The proposed design focuses on identity credentials used by digital services. It does not replace the legal or institutional process required to establish a person's identity. Credential issuers remain responsible for verifying the information they certify. The framework assumes secure cryptographic key generation, a functioning wallet application, and a governance mechanism that defines accepted issuers and credential types.

The paper presents an architecture and evaluation plan. Performance figures, implementation results, and comparative improvements must be established through a prototype and empirical testing.

4. PROPOSED BLOCKCHAIN-ENABLED IDENTITY MANAGEMENT FRAMEWORK

4.1 Architecture Overview

The proposed framework consists of five layers: user interaction, credential management, verification, blockchain registry, and off-chain data storage.

User interaction layer: Provides a digital wallet through which users receive, store, manage, and present credentials. The wallet protects private keys and supports user approval before sharing information.

Credential management layer: Supports issuer onboarding, identity proofing, credential issuance, renewal, and revocation. Issuers digitally sign credentials and provide them to users.

Verification layer: Enables service providers to request claims, validate presentations, verify issuer signatures, and check credential status.

Blockchain registry layer: Maintains issuer public keys or references, DID-related public information, and credential-status records. Smart contracts enforce permitted registry operations.

Off-chain storage layer: Holds encrypted documents or other large data objects when such storage is required. The blockchain stores references or cryptographic commitments rather than personal documents.

The architecture separates personal data from public ledger records. This separation reduces the risk of permanently exposing sensitive information through an immutable ledger.

4.2 Identity and Credential Representation

Each participant is represented by a decentralized identifier where appropriate. A DID document can contain public-key information used to verify signatures and may include service endpoints required for resolution. The DID document must not contain sensitive personal attributes.

A verifiable credential includes:

- a credential identifier;
- the subject to whom the credential applies;
- the issuer;

- the claims or attributes;
- the issuance and expiration information;
- a cryptographic proof or signature; and
- status information or a reference to a status mechanism.

For example, an educational institution may issue a credential confirming that a person has completed a qualification. A service provider that requires proof of qualification can verify the issuer's signature and credential status without requesting the institution to reissue the credential for every access attempt.

4.3 Credential Issuance

Credential issuance begins when a user requests a credential from an authorized issuer. The issuer performs the required identity proofing or checks existing institutional records. Once the issuer is satisfied, it creates a credential containing the approved claims and signs it using its private key.

The credential is delivered to the user's wallet through a protected channel. The wallet verifies the credential's basic structure and issuer signature before storing it. The issuer may publish or update relevant public information in the blockchain registry, such as issuer key references or credential-status information.

The blockchain does not independently validate the underlying claim. It records information needed by relying services to verify the credential and its status.

4.4 Credential Presentation and Verification

When a user accesses a digital service, the service provider specifies the claims required for the transaction. The wallet displays the request to the user and allows the user to approve or reject the presentation.

The wallet creates a presentation containing the required credential information and cryptographic proof. Where supported by the credential format, the wallet can disclose selected attributes rather than the complete credential. The presentation is sent to the service provider over a secure communication channel.

The service provider then:

1. validates the presentation format;
2. verifies the proof or signature;
3. confirms that the issuer is trusted for the credential type;
4. checks validity dates and credential status;
5. evaluates whether the disclosed claims satisfy the service's policy; and
6. grants or denies access.

The service provider should avoid retaining the full presentation unless retention is necessary and lawful. Authentication and authorization decisions should be recorded using minimal metadata.

4.5 Credential Revocation

A credential may need to be revoked when it is issued in error, expires, is compromised, or no longer represents a valid status. The issuer submits a status update through an authorized registry operation. The blockchain records the update, allowing service providers to check whether the credential remains valid.

A revocation mechanism must balance transparency, privacy, and operational efficiency. Publishing a unique status record for every credential may reveal usage patterns if status checks can be linked to individuals. A deployment should therefore select a status mechanism that minimizes linkability and does not expose personal attributes.

The system must also define how quickly revocation updates become effective, how service providers refresh status information, and how the network handles temporary unavailability.

4.6 Key Recovery and Credential Renewal

Loss of a wallet device or private key can prevent a user from presenting credentials. The framework should provide a recovery process that does not allow an unauthorized party to take over an identity. Possible approaches include secure backup, multi-device authorization, recovery through a trusted institution, or multi-party recovery.

Recovery should require strong authentication and produce an auditable event. If a private key is believed to be compromised, associated credentials or identifiers may need to be rotated, suspended, or revoked. The system should distinguish between recovering access to a wallet and reissuing a credential, because the latter may require the issuer to repeat identity checks.

4.7 Smart Contract Functions

Smart contracts can enforce rules for writing and updating registry records. The proposed framework may include the following functions:

- registerIssuer: records an authorized issuer or its public-key reference;
- updateIssuerKey: updates issuer key information under defined authorization rules;
- publishCredentialStatus: publishes a credential-status update;
- resolveRegistryRecord: returns public registry information; and
- recordAuditEvent: records a minimal event for authorized auditing.

These functions are illustrative. A production implementation must define access controls, transaction validation, upgrade procedures, and governance responsibilities. Smart contracts should not store identity documents, passwords, private keys, or unnecessary personal attributes.

4.8 End-to-End Workflow

The complete workflow is summarized below.

1. The user requests a credential from an authorized issuer.
2. The issuer verifies the required claims and signs the credential.

3. The credential is delivered to the user's wallet.
4. The user requests access to a digital service.
5. The service provider sends a claim request to the wallet.
6. The user approves the requested disclosure.
7. The wallet generates a presentation and sends it to the service provider.
8. The service provider verifies the proof, issuer, validity, and status.
9. The service provider applies its access policy and returns a decision.
10. If necessary, the issuer later updates the credential status through the registry.

This workflow supports repeated use of credentials while keeping identity claims under the user's control, subject to the trust and governance model.

5. SECURITY AND PRIVACY ANALYSIS

5.1 Threat Model

The framework considers attackers who may attempt to impersonate a user, forge or alter credentials, replay an earlier presentation, compromise a wallet, misuse issuer privileges, or exploit the blockchain network. It also considers privacy risks caused by excessive disclosure, correlation of transactions, and exposure of personal information through public records.

The threat analysis assumes that standard cryptographic algorithms are correctly implemented and that private keys are not directly exposed. These assumptions must be tested and reviewed during implementation.

5.2 Credential Forgery and Tampering

An attacker may attempt to alter credential attributes or create a false credential. Digital signatures allow the service provider to verify whether the credential was signed by the claimed issuer and whether its contents have changed. This protection depends on secure issuer key management and correct verification by the service provider.

5.3 Impersonation and Replay

An attacker may attempt to present another user's credential or reuse a previously captured presentation. The framework can reduce these risks by requiring proof of control over the relevant key and binding presentations to a service-specific request, nonce, or session. Service providers should reject presentations that are expired, duplicated, or not bound to the current transaction.

5.4 Compromised Issuers

A trusted issuer may be compromised or may issue incorrect credentials. Blockchain records can help make issuer information and status changes auditable, but they cannot guarantee that an issuer's claims are accurate. Governance must define how issuers are approved, suspended, audited, and removed. Services should also be able to reject credentials from issuers that are no longer trusted.

5.5 Privacy and Correlation

A public ledger can create privacy risks if identity-related records contain personal data or if repeated transactions can be linked. The framework therefore avoids storing personal attributes on-chain and recommends minimal registry records. Credential presentations should disclose only the claims required for a service. Where available and appropriate, privacy-preserving presentation mechanisms should be used to reduce unnecessary disclosure.

5.6 Availability and Network Risks

A blockchain network may experience congestion, outages, or governance disputes. If every authentication requires a live ledger transaction, service availability may be reduced. The framework should support efficient read operations, carefully managed status caching, and clear rules for handling temporary registry unavailability. Caching must not allow revoked credentials to remain accepted beyond the system's defined risk tolerance.

5.7 Security Summary

The proposed framework distributes selected identity-management functions and supports cryptographic verification, but it does not eliminate all risks. Its security depends on issuer governance, key protection, wallet security, correct status handling, and the service provider's verification process. A full security claim would require formal analysis, implementation review, and testing against defined attack scenarios.

6. EVALUATION METHODOLOGY

6.1 Evaluation Objective

The proposed framework should be evaluated to determine whether it can support secure and practical identity verification across multiple digital services. The evaluation should measure performance, scalability, revocation behavior, and privacy exposure. Since this paper does not report an implemented prototype, the following section defines a reproducible evaluation plan rather than presenting experimental findings.

6.2 Prototype Configuration

A prototype may include a wallet application, a credential issuer service, two or more relying services, a credential verification module, and a permissioned blockchain network. Test credentials should use synthetic identities and non-sensitive attributes. The prototype should record the software versions, hardware configuration, network setup, transaction parameters, and test scripts.

The comparison should include at least:

- a centralized identity baseline;
- a federated identity baseline; and
- the proposed blockchain-enabled framework.

The comparison must use equivalent authentication requirements and clearly state which operations are included in each measurement.

6.3 Metrics

The evaluation should measure the following metrics.

Metric	Definition	Purpose
Authentication latency	Time from presentation submission to access decision	Measures user-facing response time
Credential verification time	Time required to validate proof, issuer, and status	Measures verification overhead
Revocation delay	Time between a revocation update and its enforcement by services	Measures status propagation
Throughput	Successful verification requests per unit time	Measures processing capacity
Resource consumption	CPU, memory, and network use during testing	Measures implementation overhead
Registry transaction cost	Cost or resource use for registry updates	Measures ledger operation overhead
Data disclosure	Number and type of attributes released per authentication	Measures privacy exposure
Availability	Successful service decisions during network disruptions	Measures resilience

6.4 Experimental Procedure

The prototype should be tested under multiple workloads, such as low, moderate, and high concurrent request volumes. Each configuration should be executed repeatedly, with warm-up runs separated from measured runs. The evaluation should report mean, median, and percentile latency, as well as variability across repeated trials.

Revocation tests should include credentials revoked before presentation, during a period of cached status information, and while the registry is temporarily unavailable. Security tests should attempt credential modification, replay, use of expired credentials, and presentation from an untrusted issuer.

The study should also document limitations. For example, a permissioned blockchain may have different performance and governance characteristics from a public network. Results from a small prototype should not be generalized to national-scale deployment without additional testing.

6.5 Research Questions

The evaluation can be guided by the following research questions:

- RQ1: How does the proposed framework's authentication latency compare with centralized and federated baselines?
- RQ2: How does the number of concurrent verification requests affect throughput and response time?

- RQ3: How quickly do credential revocation updates become enforceable across service providers?
- RQ4: To what extent does selective disclosure reduce the amount of personal information shared during authentication?
- RQ5: Which operational costs and availability risks arise from using a blockchain registry?

These questions provide a basis for empirical evaluation without assuming that the proposed architecture is more efficient or secure in every deployment.

7. DISCUSSION

7.1 Potential Benefits

The proposed framework can reduce repeated identity proofing by allowing users to present credentials issued by trusted organizations to multiple participating services. It can also improve transparency by making issuer and credential-status information verifiable. User-held credentials may provide greater control over the presentation of identity attributes, particularly when services request only a limited set of claims.

A shared registry may be useful when several organizations need to verify issuer information or credential status but do not wish to rely on a single organization's database. The design is most appropriate where there is a clear governance structure and a defined group of participating institutions.

7.2 Deployment Challenges

The framework introduces operational complexity. Organizations must agree on credential formats, issuer authorization, status mechanisms, dispute handling, and key recovery. Users must be able to manage wallets securely and understand the consequences of sharing credentials. Service providers must implement verification correctly and maintain reliable issuer trust lists.

Interoperability is also a practical concern. A credential accepted by one service may not be accepted by another if they use different schemas, trust rules, or revocation mechanisms. Common standards and governance agreements are therefore essential for cross-service adoption.

7.3 Blockchain Selection

A permissioned blockchain may be appropriate when participants are known institutions and registry updates require controlled authorization. A public blockchain may be considered when broad public resolution is necessary, but public metadata, transaction fees, and governance need to be assessed. The selection should be based on the trust model, privacy requirements, transaction volume, and operational responsibilities rather than on decentralization alone.

7.4 Limitations

This paper presents a design framework and evaluation plan rather than a completed implementation. It does not report measured latency, throughput, cost, or security-test

results. The proposed architecture also depends on trusted issuers and effective governance, and it cannot independently verify the truth of claims made by credential issuers.

Future work should implement the framework, conduct comparative experiments, and evaluate interoperability with existing digital identity standards. User studies would also be useful to examine wallet usability, consent comprehension, and recovery experiences.

8. CONCLUSION

This paper proposed a blockchain-enabled identity management framework for secure digital services. The framework combines decentralized identifiers, verifiable credentials, user-controlled wallets, cryptographic verification, and a blockchain registry for selected public identity and credential-status information. Personal identity data is kept off-chain, while service providers verify credentials and apply access policies before granting access.

The design covers credential issuance, presentation, verification, revocation, and recovery. Its threat analysis identifies risks involving credential forgery, impersonation, replay, issuer compromise, privacy leakage, and network availability. An evaluation methodology was defined to measure authentication latency, verification time, revocation delay, throughput, resource consumption, and data disclosure.

The framework offers a structured approach to identity verification across participating digital services, but its practical value must be established through implementation and empirical testing. Future work will focus on prototype development, comparative evaluation, interoperability, and user-centered security.

REFERENCES

1. Yan, Z., Zhao, X., Liu, Y., & Luo, X. (2024). Blockchain-driven decentralized identity management: An interdisciplinary review and research agenda. Information & Management, 61(7), Article 104026. https://doi.org/10.1016/j.im.2024.104026
2. Mazzocca, C., Acar, A., Uluagac, S., Montanari, R., Bellavista, P., & Conti, M. (2024). A survey on decentralized identifiers and verifiable credentials. arXiv. https://arxiv.org/abs/2402.02455
3. He, J., Ma, X., Zhang, D., & Peng, F. (2024). Supervised and revocable decentralized identity privacy protection scheme. Security and Safety, 3, Article 2024013. https://doi.org/10.1051/sands/2024013
4. Feng, Z., Li, Z., Cui, H., & Whitty, M. T. (2025). Identity management systems: A comprehensive review. Information, 16(9), Article 778. https://doi.org/10.3390/info16090778
5. Kasarla, S. R., "A test case prioritization framework for efficient regression testing in large-scale software systems," International Journal of Applied Mathematics, vol. 38, no. 2s, pp. 1648–1661. [Online]. Available: https://ijamjournal.org/ijam/publication/index.php/ijam/article/view/2305
6. Kasarla, S. R. (2025). An empirical analysis of software maintainability using code quality and complexity metrics. Journal of Information Systems Engineering and Management, 10(4). https://jisem-journal.com/index.php/journal/article/view/15365