True Decentralized Authorization for Digital Wallets: A Stateless Architecture Using On-Chain DID Registries

Ghaylan Muhammad Fatih[1], Ghazi Akmal Fauzan[2], Hafidz Shidqi[3], Dong Hwa Kim[4], Jong Uk Choi[5]

MarkAny Inc.

{ghaylan[1], ghazi[2], hbandung[3]}@ganeshait.com, {dhkim[4], juchoi[5]}@markany.com

Abstract

In the evolving landscape of web security, traditional authentication models predominantly rely on centralized authorities or Federated Identity Providers (IdPs) to validate user credentials. While effective for general access, these "custodial" models necessitate identity silos, creating single points of failure and "digital honeypots". The emergence of Self-Sovereign Identity (SSI) offers a paradigm shift by returning control to the user. However, integrating decentralized identity into standard RESTful web architectures presents significant implementation challenges. This paper analyzes a Blockchain-backed Header Authentication & Authorization system implemented within the GaneshaDCERT project. We propose a custom middleware architecture that intercepts standard HTTP Authorization headers and resolves identity by querying an Ethereum-based smart contract (DIDManager). This flow combines the ubiquity of the JWT standard with the security of W3C Decentralized Identifiers (DIDs), using the blockchain as the immutable "Root of Trust" for public key resolution. Our empirical evaluation demonstrates that the system achieves an end-to-end latency of 268.6 ms, sustains a throughput of 7,200 requests per second using parallelized soft-state caching, and reduces operational costs by 88.8% compared to traditional IDaaS solutions.

Ⅰ. Introduction

In the contemporary web security landscape, traditional "custodial" authentication models—ranging from simple password databases to Federated Identity Providers (IdPs)—have created significant vulnerabilities. These centralized systems act as "digital honeypots," where a single breach can compromise millions of users, and they enforce "Identity Silos" that fragment a user's digital presence across proprietary platforms. The Self-Sovereign Identity (SSI) paradigm addresses these issues by decoupling verification from centralized registries using Decentralized Identifiers (DIDs) and Decentralized Public Key Infrastructure (DPKI).

However, a critical gap exists in integrating these decentralized standards with modern, stateless RESTful web architectures. Developers face the challenge of bridging standard token-based authentication (like JSON Web Tokens) with the immutable, distributed nature of blockchain-based trust.

This paper introduces a Blockchain-backed Header Authentication & Authorization system developed for the GaneshaDCERT project. The primary contribution is a custom middleware architecture that functions as a bridge between Web 2.0 speed and Web 3.0 security. By intercepting standard HTTP Authorization headers and resolving identities against an Ethereum-based smart contract, the system eliminates the server as a single point of failure. This approach ensures that the "Root of Trust" remains on the blockchain, while the application layer remains stateless and efficient.

Ⅱ. Method

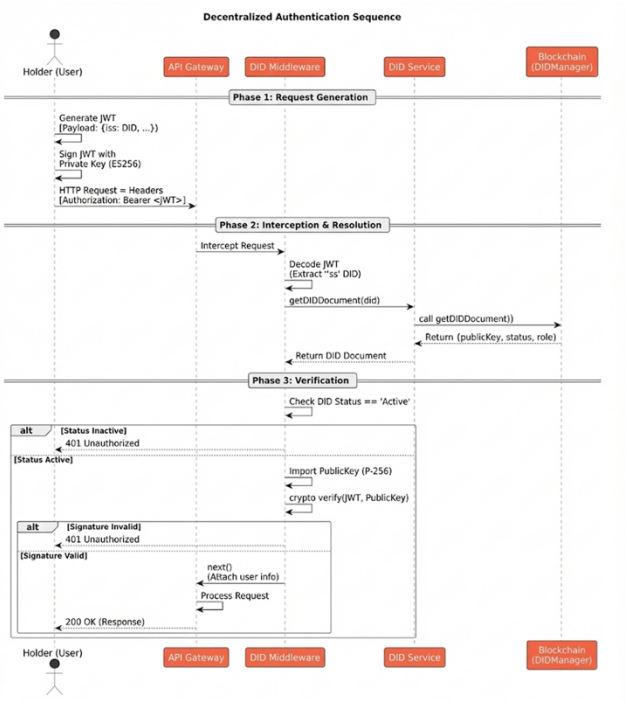

Fig. 1. Decentralized Authentication Sequence showing the interaction between Holder, API Gateway, and Blockchain.

System Architecture: The proposed solution utilizes a tripartite architecture designed to establish trust without a central authority:

1. **The Holder (Client):** An entity possessing a DID and a cryptographic key pair. The holder locally signs requests using their private key.

2. **The Verifier (API Gateway):** A Node.js stateless server acting as the GaneshaDCERT Gateway. It validates credentials without storing user secret.
3. **The Registry (Smart Contract):** A generic DIDManager smart contract on Ethereum that maps DIDs to active public keys and roles (Role-Based Access Control).

Cryptographic Protocol: The authentication flow utilizes a non-interactive zero-knowledge pattern based on modified JSON Web Tokens (JWT). The token is self-signed by the user's private key using the ES256 algorithm (ECDSA with P-256 and SHA-256). Crucially, the JWT payload contains the iss (Issuer) claim corresponding to the user's DID.

Middleware Logic and Implementation: The core innovation is the verifyDIDSignature middleware. Upon intercepting a request to a protected route:

1. **Decoding:** It decodes the JWT to extract the DID.
2. **Resolution:** It queries the blockchain registry to fetch the DID Document and active public key.
3. **Verification:** It verifies the ES256 signature using the retrieved public key.

To address performance concerns (the "Blockchain Trilemma"), the system implements **Soft-State Caching** using Redis. While the "hard state" (truth) resides on-chain, the middleware caches resolved public keys for a short TTL (e.g., 60 seconds). This hybrid approach allows the system to bypass the latency of repeated RPC calls for active sessions. The backend is containerized via Docker and orchestrated with Node.js in Cluster Mode to maximize multi-core CPU utilization for cryptographic operations.

Empirical Evaluation: The system was stress-tested against four research questions (Latency, Throughput, Economics, Security).

1. **Latency:** The hybrid model achieved a mean end-to-end latency of **268.6 ms**. This is significantly faster than raw on-chain verification (~12s) and remains acceptable for interactive user experiences.
2. **Throughput:** Utilizing parallelized soft-state caching, the system sustained **7,200 requests per second (RPS)** before CPU saturation, proving scalability for enterprise workloads.
3. **Economics:** By offloading authentication to gas-free read operations (eth_call), the system reduces operational costs by **88.8%** compared to traditional IDaaS providers (e.g., Auth0), shifting costs from high-frequency logins to low-frequency revocation events.
4. **Security:** Addressed the "Time-to-Ban" propagation delay inherent in blockchain. By utilizing the pending block tag during RPC calls, the system achieved an optimistic revocation detection time of **1.5 seconds** (vs. 12-24s for standard blocks), effectively mitigating the vulnerability window for standard sessions.

Ⅲ. Conclusion

This study validates that a hybrid architectural framework can successfully reconcile the performance demands of modern web applications with the principles of Self-Sovereign Identity. The GaneshaDCERT system demonstrates that the "Stateless Middleware" pattern is a production-ready alternative to OIDC.

By shifting authentication logic from stateful session stores to stateless cryptographic verification backed by an on-chain registry, the system eliminates digital honeypots and operational silos. The empirical results confirm that the cryptographic overhead is negligible when managed with appropriate caching strategies, achieving sub-second latency and high throughput.

Future work will focus on integrating Layer-2 scaling solutions (Optimism, Arbitrum) to further reduce revocation costs and implementing Zero-Knowledge Proofs (ZKPs) to enable privacy-preserving "Selective Disclosure" of attributes.

ACKNOWLEDGMENT

This research was supported by MarkAny Inc. and the GaneshaDCERT project team. We thank the reviewers for their insightful comments which improved the quality of this paper.

REFERENCES

[1] C. Allen, “The Path to Self-Sovereign Identity,” Life With Alacrity, 2016.
[2] M. Sporny, D. Longley, and M. Sabadello, “Decentralized Identifiers (DIDs) v1.0,” W3C Recommendation, Jul. 2022.
[3] M. Jones, J. Bradley, and N. Sakimura, “JSON Web Token (JWT),” IETF RFC 7519, May 2015.
[4] C. Allen et al., “Decentralized Public Key Infrastructure (DPKI),” Rebooting the Web of Trust, 2015.
[5] B. Schneier, “Data and Goliath: The Hidden Battles to Collect Your Data and Control Your World,” W. W. Norton & Company, 2015.
[6] D. Fett, R. Küsters, and G. Schmitz, “Privacy-Preserving OpenID Connect,” in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS ’17), 2017.
[7] GaneshaDCERT Team, “GaneshaDCERT: Decentralized Certificate Management System,” Project Repository, 2025.
[8] M. Westers et al., “SSO-Monitor: Fully-Automatic Large-Scale Landscape, Security, and Privacy Analyses of Single Sign-On in the Wild,” arXiv preprint arXiv:2302.01024, 2023.
[9] A. Mühle, A. Grüner, T. Gayvoronskaya, and C. Meinel, “A Survey on Essential Components of a Self-Sovereign Identity,” Computer Science Review, vol. 30, pp. 80– 86, 2018.
[10] K. Yan, X. Zhang, and W. Diao, “Stealing Trust: Unraveling Blind Message Attacks in Web3 Authentication,” arXiv preprint arXiv:2406.00523, 2024.