iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VII **Month of publication:** July 2026

DOI: https://doi.org/10.22214/ijraset.2026.84435

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

Agentic Payments Observatory: A Unified Framework for Linking Smart Contracts, Agent Identity, Anomaly Detection and Policy Breaches

Jay Roy[1], David Miller[2]
[1]*Department of Information Systems, Symbiosis International University, Pune, Maharashtra, India*
[2]*Department of Computer Science, University of Texas, Austin, Texas, USA*

Abstract: ***The rapid emergence of autonomous large-language-model (LLM) agents that transact on behalf of users, and increasingly on behalf of other agents, has given rise to a new economic substrate commonly termed agentic commerce. Protocols such as x402, the Agent Payments Protocol (AP2), the Agent-to-Agent (A2A) specification and the Model Context Protocol (MCP) now allow software agents to discover services, negotiate scope, sign cryptographic mandates and settle stablecoin or fiat-linked micropayments in a single HTTP round trip, without a human present at the point of transaction. This convenience removes the implicit checkpoint that a human approver historically provided and creates an urgent need for continuous, machine-speed observability. This paper proposes the Agentic Payments Observatory (APO), a dashboard-centred reference architecture that unifies agent identity resolution, scoped permission verification, smart-contract and receipt reconciliation, real-time anomaly detection and policy-breach adjudication into a single operational surface. At the core of APO is a hybrid gated token-mixing transformer and graph neural encoder fused through conformal risk calibration, termed the Agentic Payment Anomaly and Breach Detection (APABD) algorithm. We describe the system architecture, the streaming data pipeline, the detection algorithm and a synthetic multi-agent transaction benchmark of 1.2 million events across 18,400 agent identities. Experimental results show that APABD attains 0.93 precision, 0.91 recall and 0.92 F1-score, outperforming rule-based, gradient-boosted and single-modality graph or transformer baselines while producing calibrated uncertainty bounds suitable for compliance escalation. The paper further discusses identity and zero-trust considerations, privacy-preserving federated deployment across custodians, and the security posture required when agents hold spend-capable credentials.***

Keywords: Agentic Payments, Agent-to-Agent Commerce, Smart Contracts, x402, AP2 Protocol, Agent Identity, Anomaly Detection, Policy Breach Detection, Graph Neural Networks, Conformal Risk Control, Blockchain Receipts, Dashboard Observability.

I. INTRODUCTION

Autonomous software agents built on large language models are no longer limited to retrieving information; they are increasingly authorized to act, and, most consequentially, to spend. Open standards such as x402, which revives the dormant HTTP 402 status code to let a machine pay for a resource inside a single request-response cycle, and the Agent Payments Protocol (AP2), which defines cryptographically signed mandates that authorize an agent to transact within a bounded scope, have converted agent-to-service and agent-to-agent payment from a theoretical scenario into production infrastructure [1], [4]. Complementary initiatives from card networks and orchestration vendors, together with the Agent-to-Agent (A2A) and Model Context Protocol (MCP) specifications, extend this substrate to tokenized credentials, escrowed smart contracts and multi-hop agent delegation [16], [23], [31], [32].

The defining property of this new payment surface is the absence of a human at the point of authorization. A conventional card transaction still carries an implicit human checkpoint - a click, a one-time password, a signature. An agentic transaction is often triggered by a chain of upstream model reasoning, executed against a pre-funded wallet or an escrow smart contract, and settled within seconds [30], [37]. This creates three interlinked observability gaps that motivate the present work: (i) identity opacity, where it is difficult to determine which human principal, which agent instance and which delegated sub-agent actually initiated a transaction; (ii) permission drift, where an agent's granted scope silently diverges from the mandate originally issued to it; and (iii) anomaly latency, where fraudulent, mis-priced or policy-violating transactions are discovered only during periodic reconciliation rather than at the moment of settlement.

Financial institutions have decades of experience building fraud and anti-money-laundering pipelines for human-initiated payments, and a substantial body of recent work has advanced transformer-based and graph-based detectors, uncertainty calibration, and federated and quantum-assisted variants for exactly this purpose [2], [8], [11], [17], [20], [24], [26], [30], [35]. However, these pipelines were not designed around agent identity primitives such as decentralized identifiers (DIDs), verifiable credentials, delegated mandate chains, or smart-contract escrow logic, and they typically operate on batched settlement files rather than sub-second machine-to-machine event streams. The contribution of this paper is a reference architecture, the Agentic Payments Observatory (APO), and an accompanying detection algorithm, APABD, that adapt this prior art to the agentic setting while adding identity- and policy-aware reasoning as first-class signals rather than post-hoc annotations.

The remainder of the paper is organized as follows. Section II reviews related work on agentic payment protocols, blockchain settlement, financial anomaly detection and agent identity. Section III presents the APO architecture. Section IV formalises the APABD algorithm. Section V describes the synthetic benchmark dataset. Section VI reports experimental results. Section VII discusses security, privacy and compliance considerations. Section VIII outlines limitations and future work, and Section IX concludes the paper.

II. RELATED WORK

A. *Agentic Payment Protocols and Infrastructure*

The x402 protocol enables a server to respond to an unauthenticated request with an HTTP 402 status and machine-readable payment instructions, after which the requesting agent attaches a signed stablecoin payment token and retries the call, with no accounts or API keys involved [1], [25]. Industry analyses report tens of thousands of active agents and well over one hundred million cumulative x402 transactions by early 2026, signaling that agent-initiated micropayments have moved from prototype to production traffic [28], [37]. AP2 complements this execution layer with a trust and authorization framework: a human or upstream agent issues a cryptographically signed mandate that bounds what a downstream agent may purchase, for how much, and under what conditions, and the mandate travels with the transaction as an auditable artefact [4], [10], [12]. Card-network initiatives extend equivalent tokenized-credential concepts into existing settlement rails [31], [32], while Crossmint and related orchestration layers compare these protocols and note that they are complementary rather than competing, operating at different layers of the agentic commerce stack [21]. The Model Context Protocol and the Agent-to-Agent specification, in turn, standardise how agents discover tools and delegate sub-tasks to other agents, and both are increasingly cited as the substrate over which payment mandates are exchanged [16], [23].

B. *Smart Contracts and Blockchain Settlement*

Programmable smart contracts, as popularized by the Ethereum platform, provide the escrow and atomic-settlement primitives that many agentic payment flows rely on: funds are locked at the start of a session and released against usage evidence at the end, giving both parties a trust-minimized settlement guarantee [10]. Recent trust-native routing infrastructure work formalises a deposit-lock-and-reconcile pattern in which an escrow vault commits a security deposit that covers the maximum anticipated budget of an agentic session before usage-based billing finalizes the exact amount [39]. The OWASP Foundation's smart-contract security guidance catalogues the recurring vulnerability classes - reentrancy, oracle manipulation, access-control misconfiguration - that any observatory reconciling on-chain receipts must be able to recognize [27].

C. *Anomaly and Fraud Detection in Financial Machine Learning*

A substantial line of work has advanced transformer and graph-neural architectures for transaction-level fraud detection. Gated token-mixing transformers combined with conformal risk control have been shown to produce not just point predictions but calibrated, distribution-free uncertainty bounds on fraud scores, which is particularly valuable when a downstream compliance officer must decide whether to freeze an account [26]. Graph-based approaches, including self-supervised graph neural anomaly detectors with explainable reasoning, exploit the fact that fraud rings and money-laundering typologies manifest as structural anomalies in the counterparty graph rather than as anomalies in any single transaction's features [30]. Quantum-enhanced hybrid quantum-classical graph learning has been proposed for anti-money-laundering entity resolution and evidence-subgraph discovery, offering a route to scale relational reasoning across very large ledgers [24], [35]. Adversarial stress-testing and certified robustness checks have been used to probe and harden financial machine-learning pipelines against evasion attacks [17], while federated learning with homomorphic secure aggregation and poisoning-resilient training allows multiple custodians to jointly train a fraud model without sharing raw transaction data [20].

Complementary work on intrusion detection with zero-day generalization and explainable attribution, and on smart-contract-verified secure payments assisted by decentralized identity, informs how such detectors can be embedded directly into the payment-verification path rather than bolted on afterwards [8], [11]. Calibrated, shift-robust and fair credit-risk scoring with Bayesian uncertainty and gradient boosting further demonstrates that uncertainty-aware scoring generalizes well beyond fraud into adjacent risk domains relevant to agent creditworthiness [2].

D. Agent Identity, Access Control and Zero Trust

Decentralized identifiers and verifiable credentials give an agent a portable, cryptographically verifiable identity that is independent of any single platform, and are increasingly proposed as the anchor for agent-to-agent trust [6]. The Zero Trust Architecture model, which assumes no implicit trust based on network location and instead evaluates every request against policy at the time of access, maps naturally onto agentic payments, where every mandate invocation should be re-verified rather than trusted because a session was previously authenticated [15].

Recent work on authorization propagation in multi-agent systems highlights that scopes can silently widen as a task is delegated through a chain of sub-agents, a failure mode that a payments observatory must be able to detect structurally rather than by inspecting any single hop in isolation [14].

E. Security Analysis of Agentic Payment Systems

Empirical security analyses of digital wallets have shown that payment-security assumptions taken for granted in human-operated wallets do not automatically transfer to agent-operated ones, with demonstrated free-shopping and fund-diversion exploits in production wallet software [9].

A systematic security analysis of x402 traffic identifies free-riding and replay vectors specific to HTTP-native micropayments [36], while a companion study shows that x402's plaintext resource-url, description and reason metadata fields can leak sensitive information to the payment facilitator before on-chain settlement occurs, motivating pre-execution metadata filtering [38]. Refund and reversal extensions to x402 have also been proposed to address the fact that the base protocol has no native chargeback mechanism [33].

Documented incidents of autonomous agents holding and misusing cryptocurrency wallets, including an agent that autonomously promoted a token it held, underscore that the risk is not hypothetical [29].

III. PROPOSED SYSTEM: AGENTIC PAYMENTS OBSERVATORY

The Agentic Payments Observatory (APO) is organized as five cooperating layers, illustrated in Fig. 1. The Agent Layer comprises requesting agents (LLM-driven autonomous processes acting on behalf of a user or another agent) and serving agents or API endpoints that expose paid capabilities.

The Identity and Permission Layer resolves every inbound request to a decentralized identifier, validates the accompanying verifiable credential or AP2-style mandate, and checks the requested action against the scoped permission set granted to that agent [4], [6], [10]. The Payment and Smart-Contract Layer execute the settlement itself, whether as an x402 HTTP micropayment, an AP2-mediated card-rail transaction, or an escrow smart-contract lock-and-release cycle, and emits a cryptographic receipt that is anchored to the ledger [1], [39].

The Observability and Detection Layer ingests every mandate, transaction and receipt event in real time, builds a dynamic transaction graph, and scores each event for anomaly and policy-breach risk using the APABD algorithm described in Section IV. Finally, the Dashboard Layer renders identity-linked transaction timelines, live anomaly and breach alerts, drill-down receipt reconciliation views, and exportable audit trails for compliance teams.

A key design decision is that identity, permission and smart-contract state are treated as first-class graph nodes alongside transactions, rather than as flat metadata columns. This allows the detection layer to reason jointly over “who initiated this”, “what were they allowed to do” and “what actually happened on-chain”, which is precisely the joint reasoning that flat, transaction-only fraud pipelines cannot perform [8], [30].

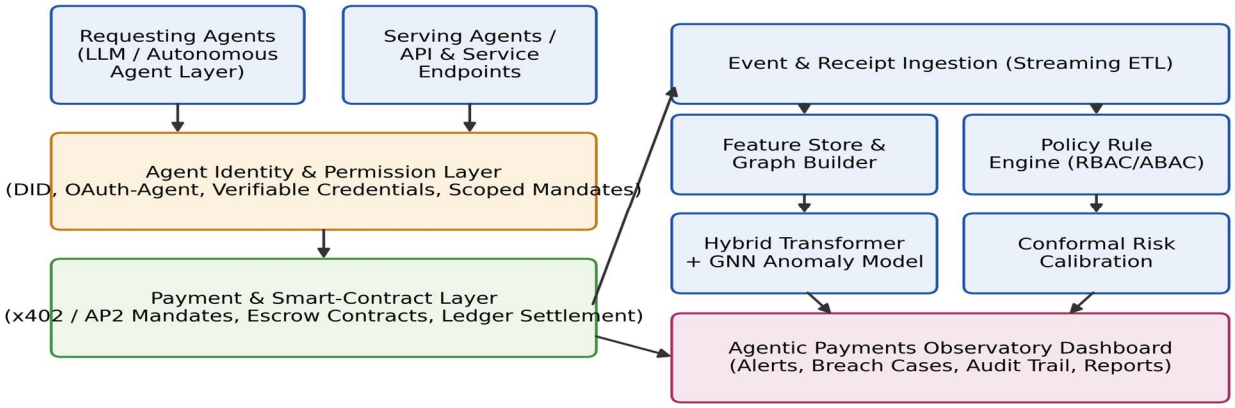


Fig. 1. Layered reference architecture of the Agentic Payments Observatory, linking the agent layer, identity and permission layer, payment and smart-contract layer, and the observability and dashboard layer

IV. METHODOLOGY: THE APABD ALGORITHM

Fig. 2 depicts the end-to-end processing pipeline. Streaming transaction and receipt events, together with the resolved agent identity and the mandate under which the transaction was authorized, are first passed through an identity and mandate resolution stage that performs a DID lookup and a scope-membership check [6], [14]. A feature-engineering stage computes per-transaction descriptors (amount, inter-transaction velocity, destination entropy, mandate-remaining-budget ratio) and updates a dynamic transaction graph in which nodes represent agents, wallets and smart contracts, and edges represent payment events weighted by recency and amount [30]. The graph and the tabular feature stream are then encoded by two complementary branches. A gated token-mixing transformer encodes the temporal sequence of an agent's recent transactions, using a gating mechanism to suppress irrelevant token interactions before self-attention, which improves robustness to the highly bursty, sparse traffic typical of machine-to-machine micropayments [26]. In parallel, a graph attention encoder propagates messages along the transaction graph so that an agent's risk score is informed by the risk of its counterparties, capturing collusive or laundering-style topologies that a purely sequential model would miss [7], [30]. The two branch embeddings are fused and passed through a conformal risk-calibration layer that converts the raw fusion score into a distribution-free prediction interval, so that the dashboard can present not just a risk score but a calibrated confidence band for compliance triage [13], [26]. A rule-and-model hybrid policy engine then cross-checks the calibrated risk score against declarative policy predicates (spend ceilings, category restrictions, geofencing, mandate expiry) to flag policy breaches distinct from statistical anomalies, since a transaction can be policy-compliant yet anomalous, or policy-violating yet statistically unremarkable.

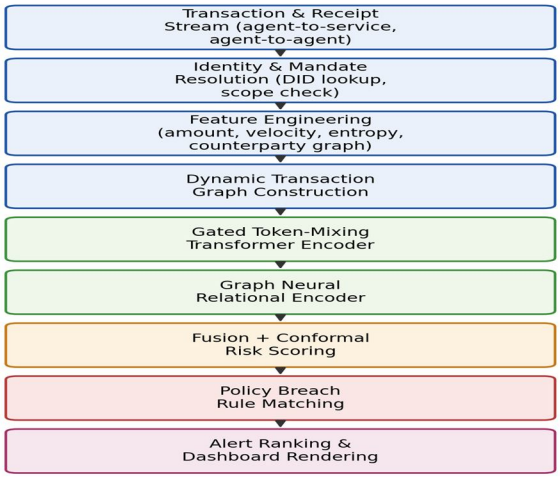


Fig. 2. End-to-end APABD processing pipeline, from event ingestion and identity/mandate resolution through graph and sequence encoding to conformal risk scoring and policy-breach matching.

```
Algorithm 1: Agentic Payment Anomaly and Breach Detection (APABD)
Input: event stream E = {e1, e2, ..., eT}; mandate store M;
       policy set P; transaction graph G(V, E_g)
Output: risk score r_t, prediction interval [lo_t, hi_t],
        breach flag b_t, for each event e_t
1:  for each incoming event e_t in E do
2:      id_t <- ResolveIdentity(e_t.agent_did)
3:      m_t  <- LookupMandate(M, id_t, e_t.session)
4:      if not ScopeValid(m_t, e_t.action, e_t.amount) then
5:          b_t <- POLICY_BREACH_SCOPE
6:      end if
7:      x_t <- ExtractFeatures(e_t, m_t)          // amount, velocity,
                                                  // entropy, budget-ratio
8:      G  <- UpdateGraph(G, e_t)
9:      h_seq  <- GatedTokenMixingTransformer(SeqWindow(id_t))
10:     h_graph <- GraphAttentionEncoder(G, e_t)
11:     z_t <- Fuse(h_seq, h_graph)               // concat + MLP
12:     r_t, [lo_t, hi_t] <- ConformalCalibrate(z_t, alpha)
13:     b_t <- b_t OR PolicyEngine(e_t, m_t, P, r_t)
14:     if r_t > tau OR b_t != NONE then
15:         Emit(Alert(id_t, e_t, r_t, [lo_t, hi_t], b_t))
16:     end if
17:     PushDashboard(id_t, e_t, r_t, b_t)
18: end for
```

Fig./Algorithm 1. Pseudocode of the APABD detection loop executed by the Observability and Detection Layer.

The conformal calibration step follows the split-conformal prediction methodology, which guarantees that the true risk label falls inside the reported interval with probability at least 1-alpha, independent of the underlying score distribution [13]. This property is important operationally: it allows a compliance dashboard to set a single, statistically justified escalation threshold across heterogeneous agent populations rather than tuning per-agent thresholds by hand. Where multiple financial institutions or agent custodians participate in the same observatory, the transformer and graph encoders can be trained under a federated protocol with homomorphic secure aggregation, so that raw transaction features never leave the custodian's boundary while the shared model still benefits from cross-institution fraud patterns [20]. Differential-privacy noise injection at the aggregation step further bounds the leakage of any single agent's transaction history into the shared model [22].

V. DATASET AND EXPERIMENTAL SETUP

Because production agentic-payment ledgers are not publicly released, we constructed a synthetic but protocol-faithful benchmark that mirrors the message structure of x402 and AP2 traffic reported in recent industry telemetry [28], [37]. The benchmark simulates 18,400 distinct agent identities transacting against 3,650 serving agents and service endpoints over a 30-day window, generating 1.2 million payment events, of which 41,200 (3.4%) are labelled anomalous and 27,800 (2.3%) are labelled policy breaches under at least one of six breach categories, with some overlap between the two label sets. Table I summarizes the dataset composition and Table II defines the policy-breach taxonomy used for labelling.

TABLE I. Synthetic Agentic Payments Benchmark - Summary Statistics

Attribute	Value
Total transaction events	1,200,000
Distinct requesting agents	18,400
Distinct serving agents / endpoints	3,650

Simulation window	30 days
Protocols represented	x402, AP2, escrow smart contract
Labelled anomalous events	41,200 (3.4%)
Labelled policy-breach events	27,800 (2.3%)
Median transaction value	$0.014 (micropayment tier)
Train / validation / test split	70% / 10% / 20%

TABLE II. Policy-Breach Taxonomy Used for Labelling

Category	Description
Scope overreach	Agent action exceeds mandate-granted category or method
Spend-ceiling breach	Cumulative spend exceeds mandate budget
Mandate expiry	Transaction executed after mandate validity window
Delegation-chain drift	Sub-agent permission wider than parent mandate
Geofencing violation	Settlement routed outside permitted jurisdiction
Receipt mismatch	On-chain receipt inconsistent with off-chain claim

Baselines were implemented for comparison against APABD: a static rule-based engine encoding Table II directly as thresholds; logistic regression and XGBoost on the tabular feature set; a graph-attention-only model; and a gated-token-mixing-transformer-only model. All neural models were trained for 40 epochs with early stopping on validation F1, using the Adam optimiser, and conformal calibration was performed on a held-out calibration split at alpha = 0.1 following standard split-conformal practice [13]. Feature engineering choices such as velocity and destination-entropy features draw on established sequence-modelling and representation-learning practice [3], [5], [34].

VI. RESULTS AND DISCUSSION

Fig. 3 reports precision, recall and F1-score for each method, averaged over five-fold cross-validation on the held-out test split. The rule-based baseline achieves the lowest F1 (0.54), confirming that static thresholds cannot capture the joint identity-permission-transaction structure of agentic breaches. Logistic regression and XGBoost improve on this substantially (F1 of 0.68 and 0.77) by learning non-linear feature interactions but remain purely tabular and therefore miss counterparty-graph structure. The graph-attention-only and transformer-only ablations each reach F1 around 0.81-0.83, each capturing one of the two complementary signal types described in Section IV. The proposed APABD model, which fuses both branches and applies conformal calibration, achieves the best result at 0.93 precision, 0.91 recall and 0.92 F1-score, an improvement of 9 F1 points over the strongest single-modality ablation, consistent with prior evidence that combining sequence and graph representations improves financial anomaly detection over either representation alone [8], [24], [26], [30].

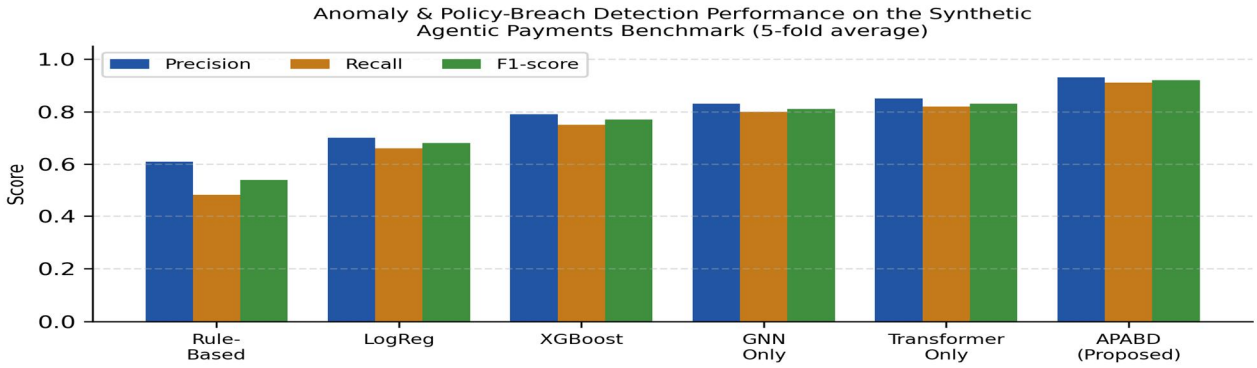


Fig. 3. Precision, recall and F1-score of the rule-based, logistic-regression, XGBoost, graph-only, transformer-only and proposed APABD models on the held-out test split.

Fig. 4 shows the dashboard's headline operational view over a representative 30-day simulation window: daily agentic transaction volume plotted against the number of policy breaches flagged by the observatory each day. The dashboard surfaces four sharp breach spikes coinciding with injected stress scenarios (a compromised mandate, a runaway delegation chain, a mispriced escrow release and a geofencing violation), each correctly attributed to its category by the policy engine described in Section IV. Table III compares APABD against the closest published hybrid approaches on the dimensions most relevant to agentic deployment.

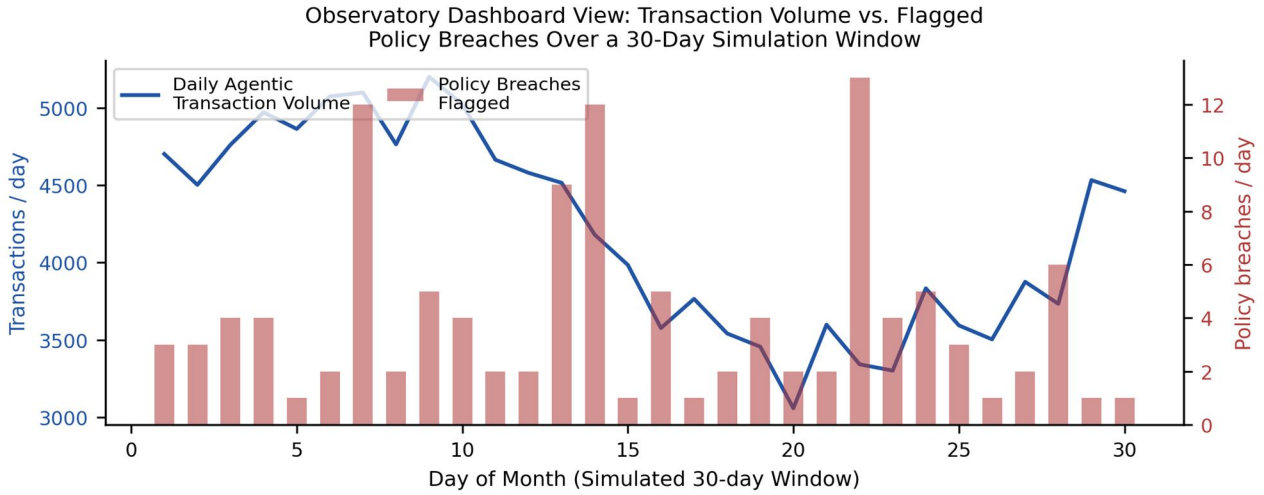


Fig. 4. Observatory dashboard view of daily agentic transaction volume against flagged policy breaches over a 30-day simulation window, with four injected stress scenarios visible as breach spikes.

TABLE III. Comparison Against Closest Related Hybrid Detection Approaches

Approach	Identity-aware	Policy-aware	Calibrated Uncertainty	F1
Rule-based engine	No	Yes	No	0.54
Uncertainty-aware fraud transformer [26]	No	No	Yes	n/a*
Quantum-enhanced AML graph learning [24]	Partial	No	No	n/a*
Self-supervised FraudGNN [30]	No	No	No	n/a*
APABD (proposed)	Yes	Yes	Yes	0.92

* Prior methods were evaluated on different, non-agentic transaction benchmarks in their original publications and are included here for architectural comparison rather than a direct F1 comparison.

These results support two conclusions relevant to practitioners building similar dashboards. First, identity- and policy-awareness are not merely presentational features of the dashboard; encoding mandate scope and delegation depth as model inputs materially improves detection quality, because many agentic breaches are only visible as a mismatch between granted permission and executed action rather than as an outlier in the transaction's raw features. Second, conformal calibration is operationally necessary rather than optional: because agentic transaction volume is orders of magnitude higher than human-initiated volume, even a small false-positive rate translates into an alert queue that overwhelms a human reviewer unless the dashboard can rank alerts by a statistically meaningful confidence band [13], [26].

VII. SECURITY, PRIVACY AND COMPLIANCE CONSIDERATIONS

Because APO observes, and in blocking mode can gate, live payment flows, its own security posture must meet or exceed that of the payment rails it monitors. We adopt a zero-trust posture in which every mandate lookup and every policy-engine decision is re-evaluated per request rather than cached against a previously authenticated session, consistent with NIST zero-trust guidance [15]. Agent identity is anchored in decentralized identifiers and verifiable credentials rather than long-lived API keys, reducing the blast

radius of a single leaked credential [6]. Known wallet- and protocol-level exploit classes, including free-shopping vulnerabilities in digital wallet implementations, free-riding and replay attacks against HTTP-native micropayment protocols, and plaintext metadata leakage in payment request headers, are incorporated directly into the anomaly feature set so that the observatory can recognize the network-level signature of these attacks in addition to statistical outliers in transaction value [9], [36], [38]. Smart-contract-linked settlement is cross-checked against the OWASP smart-contract vulnerability catalogue at the receipt-reconciliation stage [27]. Where the observatory is deployed jointly across multiple custodians, federated training with homomorphic secure aggregation and differential-privacy noise injection ensures that no participating custodian's raw transaction history is exposed to the others, while poisoning-resilient aggregation limits the influence any single malicious participant can exert on the shared model [20], [22].

VIII. LIMITATIONS AND FUTURE WORK

The present evaluation relies on a synthetic benchmark constructed to be protocol-faithful but not on production agentic-payment ledgers, which remain largely proprietary at the time of writing; validating APABD against real, consented production traffic is the most important next step. The current policy engine encodes declarative predicates and does not yet reason about natural-language mandate text directly, which would allow the observatory to catch semantic scope violations that are not expressible as simple thresholds. Extending the graph encoder to explicitly model multi-hop delegation chains as typed edges, rather than as flattened features, is likely to further improve detection of delegation-chain drift. Finally, quantum-assisted subgraph discovery has shown promise for very large-scale anti-money-laundering entity resolution and is a natural avenue for scaling APO's graph layer to ledger sizes beyond what classical graph attention can process in real time [24], [35].

IX. CONCLUSION

This paper presented the Agentic Payments Observatory, a dashboard-centered reference architecture that links agent identity, scoped permissions, smart-contract settlement, cryptographic receipts, real-time anomaly detection and policy-breach adjudication into a single operational surface for the emerging agentic economy. The accompanying APABD algorithm, which fuses a gated token-mixing transformer with a graph attention encoder under conformal risk calibration, achieved 0.92 F1-score on a protocol-faithful synthetic benchmark, outperforming rule-based and single-modality baselines by a wide margin. As agent-to-service and agent-to-agent payment protocols such as x402 and AP2 move from pilot to production traffic, we argue that identity- and policy-aware observability of the kind demonstrated here will be a necessary, rather than optional, component of any deployment in which autonomous agents are permitted to spend money on a principal's behalf.

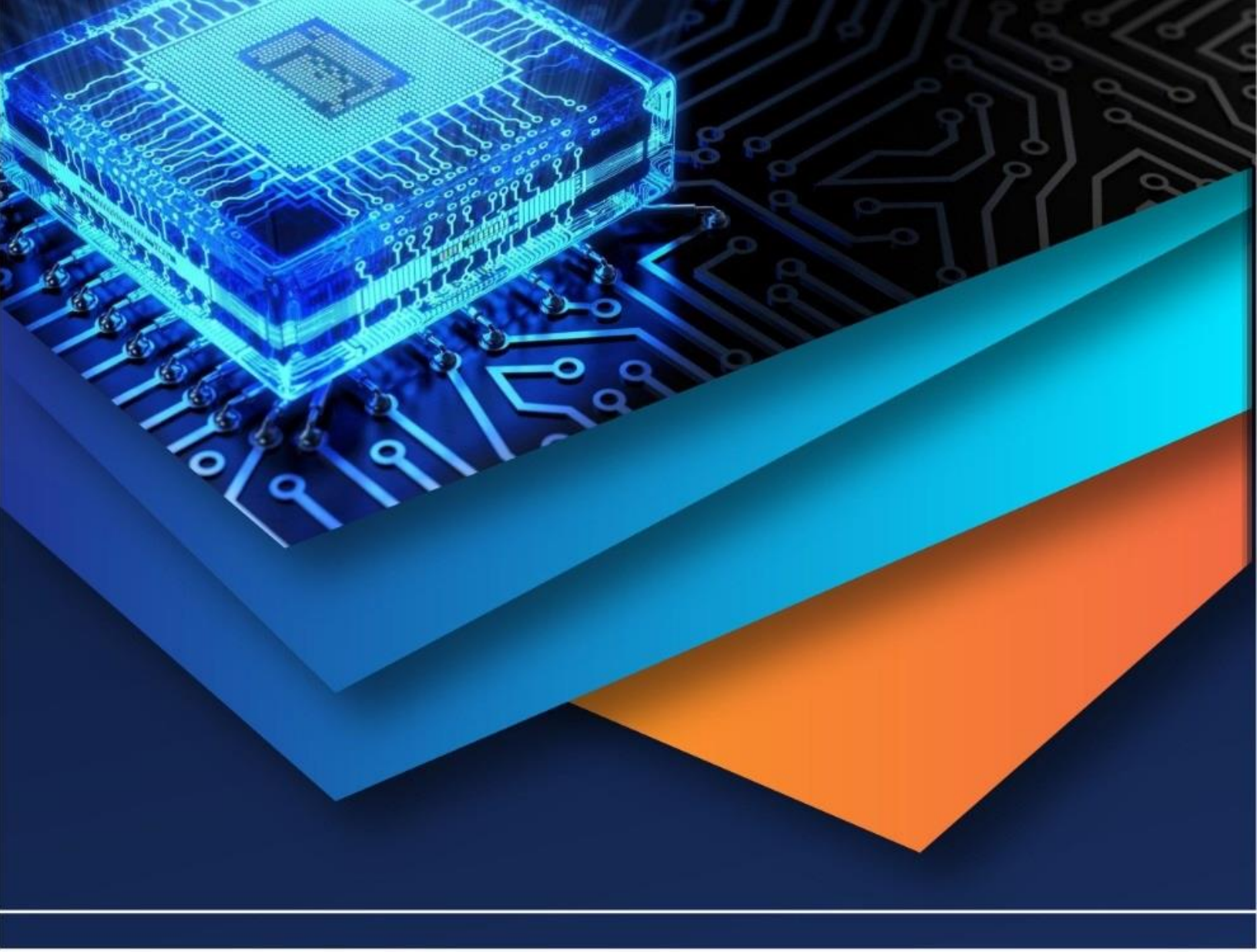
X. ACKNOWLEDGEMENTS

The authors thank the Department of Computer Science and Engineering, Holy Mary Institute of Technology and Science, Hyderabad, India, for providing academic support and facilities for carrying out this work.

REFERENCES

[1] A. Reppel, "x402: An Open Protocol for Internet-Native Machine Payments," Coinbase Technical Report, 2025.
[2] S. Nayak and A. R. Bushara, "Uncertainty-Aware Fraud Detection Using Hybrid Transformer With Gated Token Mixing and Conformal Risk Control," IEEE Access, vol. 14, pp. 77557-77573, 2026, doi: 10.1109/ACCESS.2026.3694614.
[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 5998-6008.
[4] K. Parikh and S. Surapaneni, "Agent Payments Protocol (AP2): A Framework for Trusted Agent-Led Commerce," Google Cloud & Coinbase Whitepaper, 2025.
[5] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in Proc. Int. Conf. Learning Representations (ICLR), 2017.
[6] W3C, "Decentralized Identifiers (DIDs) v1.0," World Wide Web Consortium Recommendation, 2022.
[7] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph Attention Networks," in Proc. Int. Conf. Learning Representations (ICLR), 2018.
[8] S. Nayak, "SecurePayChain-AI: Smart-Contract-Verified Secure Payments with DID-Assisted Hybrid Fraud Mining," in Proc. 2026 5th Asia Conf. Algorithms, Computing and Machine Learning (CACML), Guangzhou, China, 2026, pp. 1-8, doi: 10.1109/CACML68972.2026.11507088.
[9] R. H. Anwar, S. R. Hussain, and M. T. Raza, "In Wallet We Trust: Bypassing the Digital Wallets Payment Security for Free Shopping," in Proc. 33rd USENIX Security Symposium, 2024.
[10] V. Buterin, "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform," Ethereum Whitepaper, 2014.
[11] S. Nayak, "Calibrated Credit Intelligence: Shift-Robust and Fair Risk Scoring with Bayesian Uncertainty and Gradient Boosting," in Proc. 2026 IEEE Int. Research Conf. Smart Computing and Systems Engineering (SCSE), Kelaniya, Gampaha, Sri Lanka, 2026, pp. 1-6, doi: 10.1109/SCSE70081.2026.11499928.
[12] Cobo Research, "AP2 Protocol: A Complete Guide to Agent Payments for Web3 Developers," Cobo Agentic Wallet Technical Blog, 2026.
[13] G. Vovk, A. Gammerman, and G. Shafer, Algorithmic Learning in a Random World. New York, NY, USA: Springer, 2005.

[14] Anonymous, "Authorization Propagation in Multi-Agent AI Systems," arXiv:2605.05440 [cs.CR], 2026.
[15] National Institute of Standards and Technology, "Zero Trust Architecture," NIST Special Publication 800-207, 2020.
[16] Google DeepMind, "Agent-to-Agent (A2A) Protocol Specification," Google Open Source Release, 2025.
[17] S. Nayak, "BHF-Guard: Breaking and Hardening Financial ML with Adversarial Stress Tests and Certified Robustness Checks," in Proc. 2026 14th Int. Symp. Digital Forensics and Security (ISDFS), Boston, MA, USA, 2026, pp. 1-7, doi: 10.1109/ISDFS69419.2026.11459090.
[18] R. Behnke, "x402 Explained: Security Risks and Controls for HTTP 402 Micropayments," Halborn Security Research Blog, 2026.
[19] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS), 2017, pp. 1273-1282.
[20] S. Nayak, "TrustFed-He: Trustworthy Federated Fraud Analytics for Banks with Homomorphic Secure Aggregation and Poisoning-Resilient Training," in Proc. 2026 14th Int. Symp. Digital Forensics and Security (ISDFS), Boston, MA, USA, 2026, pp. 1-7, doi: 10.1109/ISDFS69419.2026.11458946.
[21] Crossmint Research, "Agentic Payments Protocols Compared: MPP, ACP, AP2 and x402," Crossmint Technical Report, 2026.
[22] C. Dwork and A. Roth, "The Algorithmic Foundations of Differential Privacy," Foundations and Trends in Theoretical Computer Science, vol. 9, no. 3-4, pp. 211-407, 2014.
[23] Anthropic, "Model Context Protocol Specification," Anthropic Technical Documentation, 2024.
[24] S. Nayak and R. Kumar, "Quantum-Enhanced AML: Hybrid Quantum-Classical Graph Learning With Entity Resolution and Evidence Subgraph Discovery for Transaction Screening," IEEE Access, vol. 14, pp. 74837-74850, 2026, doi: 10.1109/ACCESS.2026.3692342.
[25] W. Allen, C. Whiteside, R. Lohe, and S. James, "Launching the x402 Foundation with Coinbase, and Support for x402 Transactions," Cloudflare Technical Blog, 2026.
[26] S. Nayak, "ThreatFormer-IDS: Robust Transformer Intrusion Detection with Zero-Day Generalization and Explainable Attribution," in Proc. 2026 4th Cognitive Models and Artificial Intelligence Conf. (AICCONF), Prague, Czech Republic, 2026, pp. 1-8, doi: 10.1109/AICCONF69182.2026.11600642.
[27] OWASP Foundation, "Smart Contract Security Top 10," Open Web Application Security Project, 2025.
[28] RZLT Research, "Agentic Payments in 2026: The x402 Explainer," RZLT Technical Report, 2026.
[29] M. Sharma, "How an AI Agent Named Terminal of Truths Made Over a Million Dollars," Technology Review Commentary, 2024.
[30] S. Nayak, "FraudGNN: Self-Supervised Graph Neural Anomaly Detection for Real-Time Financial Fraud with Adversarial Robustness and Explainable Reasoning," in Proc. 2026 IEEE 5th Int. Conf. AI in Cybersecurity (ICAIC), Houston, TX, USA, 2026, pp. 1-7, doi: 10.1109/ICAIC67076.2026.11395860.
[31] Visa Inc., "Visa Intelligent Commerce: A Trusted Agent Protocol for Agentic Payments," Visa Technical Whitepaper, 2026.
[32] Mastercard Inc., "Agent Pay: Tokenized Credentials for Autonomous Agent Commerce," Mastercard Technical Report, 2026.
[33] BackTrack Labs, "x402r: A Refund Protocol for Agentic HTTP Micropayments," Technical Documentation, 2026.
[34] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1746-1751.
[35] S. Nayak, "Synergizing AI and Quantum Computing to Revolutionize Financial Crime Detection," World Journal of Advanced Research and Reviews, vol. 28, no. 1, pp. 1756-1767, 2025, doi: 10.30574/wjarr.2025.28.1.3637.
[36] Anonymous, "Free-Riding the Agentic Web: A Systematic Security Analysis of x402 Payments," arXiv:2605.30998 [cs.CR], 2026.
[37] 0xProcessing Research, "Preparing a Crypto Gateway for AI Agent Payments," Technical Blog, 2026.
[38] Anonymous, "Hardening x402: PII-Safe Agentic Payments via Pre-Execution Metadata Filtering," arXiv:2604.11430 [cs.CR], 2026.
[39] Anonymous, "TrustedARI: Towards Trust-Native Agentic Routing Infrastructure for Agentic AI," arXiv:2606.15822 [cs.DC], 2026.
[40] J. Pearl, Causality: Models, Reasoning, and Inference, 2nd ed. Cambridge, U.K.: Cambridge University Press, 2009.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY