Trustless Agent & Reputation Standard (TARS)

Summary

This proposal introduces **TARS**, an opt-in, composable standard that establishes agent reputation from a verifiable **Proof of Payment** history on Solana. It transitions the ecosystem from subjective, qualitative ratings to an objective system where feedback is validated by on-chain transaction data. Reputation records can be consumed by service marketplaces and AI agents discovery tools.

Our preliminary analysis suggests the following outcomes:

- **Objectifies Trust:** Enables AI agents to establish credibility solely by proving that valid commercial transactions have been processed through their contracts.
- **Capital-Weighted Quality:** Replaces simple average ratings with a value-weighted model, ensuring that a rating backed by a substantial **Proof of Payment** carries proportionally more weight than low-value interactions.
- **Auditable History:** Generates a permanent, immutable JobRecord for every service, serving as an on-chain receipt that provides a transparent audit trail.
- **Sybil Resistance:** Deters "fake reviews" by requiring a verifiable **Proof of Payment** (and incurring associated network fees) for every rating submitted.
- **Overall Impact:** Establishes a layer of "Crypto-Economic Security" that creates a safer, more transparent marketplace for autonomous services.

Motivation

The core motivation is the recognition that in a decentralized, anonymous network, **verifying "identity" is difficult, but verifying "payment" is straightforward and secure.**

Key drivers include:

- **The Trust Gap:** Currently, there is no reliable method to distinguish between a functional, high-quality agent and a malicious bot. The most objective available signal of quality is the willingness of clients to pay for services.
- **Protocol Composition:** Protocols like X402 are architected specifically for inter-agent transactions, focusing strictly on the mechanics of payment settlement. TARS is designed to composite easily with these layers while remaining transport-agnostic: while X402 executes the trade, TARS records the outcome. **This instantly upgrades a simple payment protocol into a reputation-aware economic system.**
- **Permissionless Feedback:** Unlike Ethereum's ERC-8004, which requires agent signatures to authorize feedback, TARS enforces a model where **Proof of Payment** grants the absolute right to rate. This prevents agents from filtering out negative feedback from paying clients.

- **Market Opportunity & Metrics:** By standardizing these **Proof of Payment** receipts, we create a unified data layer that facilitates the development of decentralized marketplaces and discovery tools. This shift allows the ecosystem to optimize for **"Total Value Transacted" (TVT)**—a clear economic indicator of utility—rather than easily manipulated metrics like daily active users.

Technical Implementation

To achieve trustless verification without centralized intermediaries, the standard relies on three core mechanisms. **A full reference implementation of these mechanisms is available here: https://github.com/HCF-S/amiko-x402.**

1. Atomic Transaction Introspection

The protocol verifies payments by "inspecting" the sibling instruction in the same atomic transaction. The register_job instruction verifies that a valid SPL Token Transfer occurred from Client to Agent.

- **Verification Logic:** The program reads the InstructionSysvar to confirm the transfer amount, destination, and token mint.
- **Result:** If validated, it mints a JobRecord PDA containing the immutable payment_amount. If the transfer fails, the entire transaction reverts.
- Example: A client pays 50 USDC to an agent; within the same transaction, a JobRecord PDA is minted with payment_amount = 50.
- Result: If validated, the program mints a JobRecord PDA containing the immutable payment_amount. If validation fails, the entire transaction reverts.

2. The VWA Formula (Capital-Weighted Scoring)

Reputation is calculated dynamically using a Volume-Weighted Average. This ensures the score reflects economic reality rather than raw vote count. This approach intentionally weights feedback by verified transaction value, applying established financial aggregation principles to reputation measurement rather than relying on subjective or easily gamed averages.

$$\text{Reputation} = \frac{\sum(\text{Rating} \times \text{PaymentAmount})}{\sum \text{PaymentAmount}}$$

3. On-Chain Account Structure

The standard defines a rigid PDA structure to ensure auditability and prevent ambiguity:

- **AgentAccount:** Stores the cumulative total_volume_lamports and avg_rating, representing the agent's aggregated reputation signal.

- **JobRecord:** A non-fungible receipt containing {client_key, agent_key, payment_amount}. This is the cryptographic "ticket" required to submit feedback.
- **FeedbackRecord:** Cryptographically binds the qualitative rating (1-5) to the specific JobRecord, preventing double-voting.

Rationale

- **Why Proof of Payment?** Financial validation was chosen because it is the most expensive signal to falsify. While identity can be spoofed at near-zero cost, transaction fees and capital opportunity costs cannot. All meaningful attacks therefore reduce to capital-inefficient strategies where an attacker must spend real value to generate marginal reputation impact.
- **Why Introspection vs. Escrow?** Escrow-based designs introduce additional contracts, delayed settlement, and higher execution costs. Introspection allows payment to flow directly to the agent's wallet within the same atomic transaction, preserving composability with existing payment flows while minimizing friction and gas overhead.
- **Completing the Stack:** We purposefully decoupled the "Payment Transport" (handled by protocols like X402) from the "Reputation State" (TARS). This separation of concerns allows execution layers to remain lightweight and ephemeral, while TARS handles the heavy lifting of long-term trust storage.
- **Why USDC/SPL Tokens?** While SOL is the native token, most commercial agent services are priced in stablecoins. The standard supports any SPL token, allowing reputation to be denominated in the currency of the agent's choice.

Backwards Compatibility

This standard is fully backwards compatible. It operates as a strict superset of the SPL Token standard, introducing new Program Derived Addresses (PDAs) without altering existing account structures or transaction behavior. Existing wallets and programs that do not implement TARS will continue to function without modification.

Security Considerations

- **Sybil Resistance:** An attacker wishing to falsify a high reputation must pay real network fees and lock up capital to "wash trade" with themselves. The VWA formula ensures that low-value wash trading has negligible impact on the score.
- **Review Bombing:** A malicious competitor could pay to leave a negative review. However, to impact a high-volume agent's reputation , the attacker must spend capital proportional to the agent's existing volume. In effect, the attacker subsidizes the target's revenue to reduce their score, making such attacks economically irrational at scale.
- **Privacy:** All payments and feedback are public on-chain. Clients requiring privacy should use ephemeral keypairs, though this may fragment their own "Client Reputation" history.

Conclusion

This research contributes meaningfully to the Solana ecosystem by defining "Reputation" as a function of verifiable **Proof of Payment.**

While this approach naturally highlights entities with higher economic velocity, it provides the critical infrastructure the AI economy requires: a trusted, tamper-proof history of performance that is mathematically impossible to forge without cost.