PIN AI: The Coordination Layer for Personal AI

The PIN AI Team

November 2024

Abstract

PIN AI is a blockchain network designed to enable economic interactions between humans and AI agents through Personal AI, while preserving user privacy and leveraging personal data. The PIN Protocol combines Personal AI, AI Agent services, and smart contracts to create a decentralized network that supports personal data processing, intent matching, and secure agent interactions. This whitepaper introduces the key components of the PIN Protocol, including PIN Chain, the Personal Data Protocol (PDP), and the Agent Service Protocol (ASP), and outlines the economic incentives and mechanisms that enable efficient and secure AI services.

1 Introduction

PIN AI is the coordination layer for Personal AI. The goal of the PIN protocol is to enable economic interactions between humans and AI agents, mediated by Personal AI. Our goal is to enable AI developers to deliver useful AI services, such as purchasing retail items online, organizing a trip, planning financial or health-related actions over a long horizon. Today, these are not possible because cloud AI models like ChatGPT lack the necessary user context, history, and preferences. Moreover, in the current model there is high risk of privacy leakage from data-sensitive apps.

Personal AI applications necessitate access to a wide range of personal data across various Web2 platforms. To balance this need with user privacy, PIN AI is designed with a hybrid model that combines Personal AI and AI Agent Services.

- **Personal AI** operates on the user's device or private cloud, preserving privacy and efficiently maintaining full context of the user via preference embeddings that are continuously updated.

- **AI Agent Services** are specialized AI applications designed for specific tasks, which have programmable access to personal data controlled by the user and can interact in private compute environments. They can be cloud-based or privately hosted; the only requirement is that they conform to our protocol interface discussed below.

The Web2 infrastructure is not fit to offer such services. Big Tech companies have rich user context from their own applications, but they cannot utilize user data from their competitors, creating a hard wall for Personal AI. The wide range of AI Agents in service of Personal AI need a matching platform which leverages personal user context for efficient matching. This needs to be a neutral platform where users can self-custody and safely connect their personal data avoiding surveillance; users can monetize the economic value of their personal data and intents avoiding 100% extraction from Big Tech services; AI developers can access an open platform and deliver services without paying any rent [4].

The PIN Protocol is built on top of the latest improvements in blockchain scalability and offchain private computation to enable Personal AI. We designed the PIN Protocol as the economic layer for PIN AI. The main goal is to incentivize a diverse set of agent and data service providers and maintain high levels of service quality with extremely low fees. There are three main components: Personal Data Protocol, Agent Service Protocol, and PIN Chain.

2 PIN Chain

PIN Chain is a new blockchain that is tailored for Personal AI applications. It integrates several key features that make it easy for AI developers to deploy smart contract logic for their AI applications and verify inference and actions onchain. In particular, the chain supports cheap and fast execution of transactions with AI services, including conditional actions and payments, and it manages native incentives for AI services. It enables Personal AI with two main innovations:

- **Private data and compute for Personal AI**: A native Trusted Execution Environment (TEE) coprocessor supports tasks that require processing of personal data, including interactions with AI services that require personal information. Only metadata is publicly logged onchain to enable auditing, reputation, and other incentive mechanisms. We will also explore private block-building mechanisms similar to rollup-boost [3] for economically valuable transactions that generate MEV.

- **Decentralized coordination of AI services**:

Core features for the creation of a decentralized marketplace of AI services are also natively deployed and optimized:

1. **Registry for AI agents and data services**, with data logging for reputation scoring, and economic security via native staking.
2. **Efficient matching algorithm based on user preference embeddings** that allows for optimal service delivery and quality.
3. **Protocol for AI agent and Personal AI interaction** with onchain verification.

3 Personal Data Protocol (PDP)

To empower personal AI applications with rich user context while preserving privacy, the PIN protocol uses **decentralized data connectors** that securely access and process personal data from various platforms within a TEE. Data connectors facilitate seamless integration through cryptographically secure methods supported by an open-source framework that is publicly maintained and improved to adapt to new use cases.

3.1 Protocol Workflow

Data connectors operate through a secure and efficient workflow designed to protect user data.

1. **User Submission**: Users submit data fetching jobs to the mempool.

2. **Batching**: The block proposer batches multiple user requests and assigns them to data connector nodes based on a job allocation mechanism.

3. **Data Retrieval and Attestation**: Data connector nodes retrieve the user's personal data from the respective platforms within the TEE. After processing, they generate cryptographic attestation reports, confirming that operations were executed correctly, and submit these to the PIN chain.

4. **Onchain Verification**: The PIN chain verifies the attestation reports, ensuring the integrity and authenticity of the data processing without exposing sensitive information.

This protocol is built to ensure that data handling is secure and verifiable. It also accommodates user preferences for data access and privacy, integrating flexibility directly into the system's operation. In particular, we have the following properties: **privacy** — the data fetching is processed within a TEE, ensuring that the personal information remains confidential; **verifiability** — onchain verification of cryptographic proofs from data operations confirms that processing within the TEE is accurate and secure against tampering; **flexibility** — users have flexibility in how their data is accessed, with **on-demand updates** they request updates to their personal information whenever they need to refresh their data, with **scheduled fetching** they configure applications to allow data connectors to fetch data at predetermined intervals.

3.2 Staking and Job Allocation

The Staking and Job Allocation Protocol ensures a robust network of data connectors by combining a Proof of Stake (PoS) mechanism with a fair job distribution system.

1. **Staking Requirement**: To become a data connector, one must stake at least N_{dc} tokens in the beginning.

2. **Allocation Rule**: For each batched set of jobs, we randomly select an idle data connector. The probability of selecting the i-th data connector is

$$p_i = \frac{\min\{N_i, N_{\text{dc}}\}I_i}{\sum_k \min\{N_k, N_{\text{dc}}\}I_k},$$

   where N_k is the staking amount of the k-th data connector and I_k is the indicator variable showing whether the k-th data connector is idle.

3. **Fault Detection**: If a data connector does not complete the job within a predetermined number of epochs, a penalty of $S_{\text{dc}}^{\text{live}}$ will be imposed. Additionally, if a data connector fails the verification procedure, it will incur a penalty of $S_{\text{dc}}^{\text{safe}}$. If the staking amount for a data connector falls below the protocol requirement αN_{dc}, it will no longer be eligible to operate as a data connector.

3.3 Economic Incentives for PDP Participants

By aligning economic incentives with robust security measures, the Personal Data Protocol ensures a sustainable and trustworthy data connector ecosystem. The fee and payment structure for data connectors is as follows.

- **Fees**: To prevent spam, users pay a minimal fee (e.g., $0.01) per request for data connector services.

- **Payment Method**: Users are encouraged to pay fees using native tokens, which may offer cost benefits. If users choose to pay with other tokens like USDC, the fee will be slightly higher, with the additional amount directed to the treasury pool to support network development and maintenance.

- **Cost Efficiency**: The minimal fee to the data connector is sufficient to cover the operational costs of running data connectors. For example, renting a large instance on a TEE cloud provider costs less than \$0.40 per hour. If a request takes around 10 seconds to process, the cost per request is approximately \$0.001, which is well below the minimal fee.

Users are incentivized to provide useful and timely data to the network through token rewards. The total token reward to the user U is allocated based on a score assigned to each data contribution. This score, represented by d_i for each data entry, reflects the **usefulness, quality, and timeliness of the data**. Higher quality and more recent data receive higher scores, incentivizing users to provide valuable and up-to-date information. To maintain balance and prevent inflation, the reward diminishes as more data is contributed, following a decay formula.

- **Decay Formula**: The reward is halved when the cumulative score of useful data reaches a threshold D. This is governed by the decay rate, λ_{user}, defined as
$$1 - e^{-\lambda_{\text{user}} D} = \frac{1}{2}. \tag{1}$$
- **Reward Calculation**: Given a sequence of data contributions $\{d_i\}$, where each d_i represents the score of the i-th data entry, the reward for the user who provided data d_k is
$$\left(e^{-\lambda_{\text{user}} \sum_{i=1}^{k-1} d_i} - e^{-\lambda_{\text{user}} \sum_{i=1}^{k} d_i}\right) U. \tag{2}$$

To prevent users from submitting data through multiple accounts and earning multiple rewards, we implement **a robust identity verification mechanism**. The reward structure and identity verification are designed to: (1) discourage users from submitting irrelevant or low-quality information, since there is no reward for useless data; (2) incentivize users to submit their data and engage with our network as soon as possible to maximize their benefits; (3) discourage users from splitting their data across multiple accounts; (4) avoid reward for duplicate data among different accounts while maintaining user privacy. The structure of the identity verification system is as follows.

- **Unique Identifier Encryption**: Each user's unique identifiers (such as email or Amazon account) are encrypted using a common public key assigned to all data connectors.
- **Privacy**: No one possesses the corresponding private key, ensuring that encrypted identifiers cannot be decrypted and user privacy is maintained.
- **Address Linking**: The encrypted identifier is directly linked to the user's specific PIN address.
- **Duplicate Detection**: When a data submission occurs, the data connector checks if the encrypted identifier is already associated with another PIN address within the TEE.
- **Submission Rejection**: If the same encrypted identifier is detected for a different address, the submission is automatically rejected.

4 Agent Service Protocol (ASP)

The PIN Protocol is designed to provide economic coordination for different types of agent services. The goal is to bootstrap the creation and support the operation of powerful and diverse agent networks. There are two main types of agents.

- **Affiliate Agents**: These are agents that are affiliated with a real-world business (e.g., an online marketplace) or a protocol. They can be operated by the business itself or a third party. They can bid on personal intents and compete for optimal intent fulfillment given user preferences. They take a commission from the user transaction or other form of payment from the business they are affiliated with, so they require **no direct payment from the user**.
- **Independent Agents**: These are agents that interact with the personal AI to provide or request a **direct service and payment**. They may request payment when they provide a service, e.g., health counseling or investment advice, or generate a piece of digital art. They may send payment when they enlist the Personal AI to perform some service aligned with the agent's goals, e.g., a community leader AI that can allocate resources to build a following.

Our native application has many agents of the affiliate type, which are divided into categories for efficient matching. For example, an Amazon affiliate agent (AMZ in Figure 1) can help the user find the best product they are looking for via interaction with the user's Personal AI. We built initial use cases like this as a demonstration. In practice, our protocol supports a wide variety of services, based on single or multi-agent networks, and payment mechanisms as we describe in the following section.

4.1 Economic Layer for AI Agents

The main features our protocol provides to agentic application builders are:

- **Payment Mechanisms**: PIN Chain enables fast and cheap micropayments between agents. Plus, the PIN Protocol enables programmable payments, conditional on events or the AI agents proving a result. We also enable different types of payment mechanisms for the different types of economic relationships, e.g., Personal AI pays a worker agent to get some task done, gets paid by a master agent to contribute some data or work, Affiliate Agents bid in an auction for intent matching.

- **Shared Ownership**: Agents can be owned by a single entity or a collective of individuals. The PIN Chain has pre-deployed smart contracts that make it easy for the agent creator to implement mechanisms of shared ownership and, when applicable, revenue sharing or other economic incentive mechanisms.

- **Agent Reputation**: Agents have success metrics that are logged onchain [6], together with other behavior metadata, and are used to compute agent reputation scores. These can be used as input in matching and also in slashing conditions, to implement economic security and quality of service.

- **Verifiability**: Agent actions and events can be verified onchain via the verifier contract. Onchain actions that are not atomically executed can be verified by submitting a valid zk-proof; offchain actions require a zkTLS [7, 8, 9] or an intersubjective oracle. We promote the deployment of infrastructure that enables verifiability.

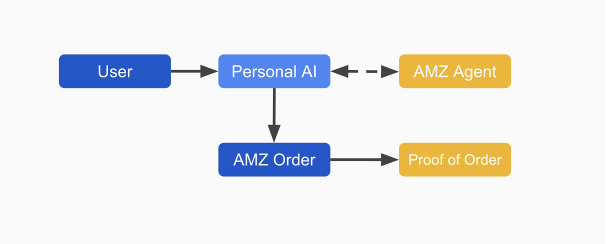

Figure 1: Agent Verification Flow

4.2 Intent Matching Protocol

When the user submits a request to their Personal AI, it can either be fulfilled immediately, or the Personal AI can compose an intent and send it to a market of External AI Agents. Agents compete to secure fulfillment rights, and the protocol matches the best agent (based on agent properties, price, and user preferences). Note that the notion of **user intent in PIN AI is much more general** than the restrictive notion of partial transaction typically used in Web3. Every Personal AI query reflects a user's desires, needs, or questions at a specific moment [5]. The Personal AI helps translate the query into a structured intent composed of a natural language prompt plus some required components, and then fulfills it either directly or by calling an agent service via the intent matching protocol.

Here we present an instance focused on interaction between Personal AI and Affiliate agents. To efficiently connect users with the most suitable AI agents, the PIN Protocol introduces a decentralized Intent Matching system. This system ensures optimal service delivery while maintaining privacy and fairness. The intent matching algorithm described here serves as an example that can be applied to a variety of agents, including affiliate agents and independent agents.

The PIN Chain has a block builder which is modified to provide intent matching functionality. The builder receives user intents, AI Agents read open intents and send bids for intents they want to serve, the builder processes intents and bids to build a full block. This process can also run in a TEE to preserve privacy of intents and bids [2], in this case the validators also need to verify the attestation report.

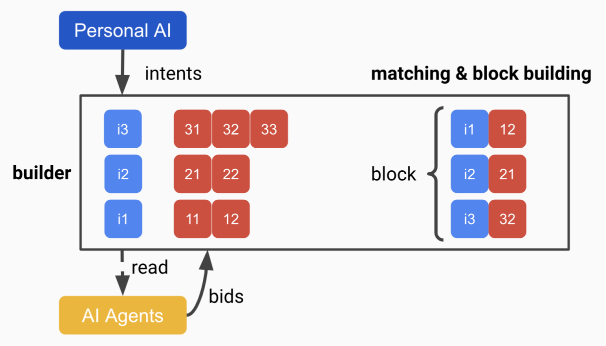

Figure 2: Intent Matching Mechanism

An intent in the PIN Protocol is a new transaction format with the following key elements:

- **Category**: Indicates the category of the intent, such as swapping agent, staking agent, or flight agent.

- **Budget**: Specifies how much the user is willing to pay for intent fulfillment to the service provider (which can be zero).

- **Payment**: An option specifying the type of payment, such as crypto or Web3.

- **Prompt**: A text or encoded prompt, for example, "Find me a health coach service to meet once per week."

- **Preferences Embedding**: Preference embeddings for the user.

The bid submitted by an agent contains:

- **Bid Amount**: A numerical value representing the bid amount.

- **Features Embedding**: An embedding vector representing the agent's features, such as previous bidding history, reputation score, or product catalog.

- **Agent Requirements**: A set of intermediate steps and information that the agent needs to complete their final task.

The intent matching algorithm uses a variant of cosine similarity between embeddings and bid amounts to evaluate and select the most suitable agent. It considers both the agents' bids and their features embeddings. For example, agents with higher reputation and competitive bids are more likely to win, promoting high-quality services at lower costs. The intent matching process follows several steps:

1. **Intent Submission**: The user submits their intent to the mempool and locks a predefined gas fee.

2. **Bid Submission**: In response to the user's intent, agents send their bids.

3. **Intent and Bid Collection**: After a specified time period has elapsed since the intent was added to the mempool, the block builder aggregates all bids and gathers features embeddings for the agents interacting with the intent, if at least one agent bids for the intent. Otherwise, the intent will be dropped from the mempool.

4. **Intent Matching**: The block builder executes an intent matching algorithm to identify agents capable of providing the relevant service to the user.

5. **On-Chain Confirmation**: The matching information is displayed on-chain.

6. **Service Provision**: The matched agent provides the necessary information or service to the user and may charge a fee.

7. **Bid Handling**: For the selected agent's bid, a specified portion is sent to the user. If the user accepts the agent's service, the remainder of the bid is returned to the agent. If not, the remaining amount is allocated to the treasury.

The key properties of the intent matching algorithm are: **verifiability** — the selection process is transparent and verifiable, ensuring that agents are chosen fairly based on predefined criteria; **reputation** —other things equal, agents with higher reputation and competitive bids are more likely to win, promoting high-quality services at lower costs; **equilibrium** — by balancing service quality and fees, the intent matching algorithm is committed to matching users with agents offering optimal services, fostering a competitive and efficient marketplace.

4.3 Agent Communication Protocol

Once the best agent service is matched, a communication channel is started between the AI Agent and the Personal AI, and they follow a secure interactive protocol to fulfill the user's intent. The protocol follows certain standards, according to parameters that are submitted at the intent-bidding phase and are recorded on the PIN Chain.

For example, the Personal AI interacts with an AI Agent to book a flight ticket. The Personal AI openly provides non-sensitive information such as the destination, dates, airlines, and seating preferences. The AI Agent processes this information and suggests suitable flight options. When proceeding with a booking, sensitive information like access tokens and payment details are securely submitted in a TEE. The AI Agent submits the selected flight details to create a booking request. The TEE then makes a secure API call to the flight booking service (e.g., Expedia) using the combined information. The entire transaction, including the handling of sensitive data and the API call, occurs within the protected enclave. This ensures that the user's confidential information remains secure and is not exposed or retained by the AI agent outside the TEE.

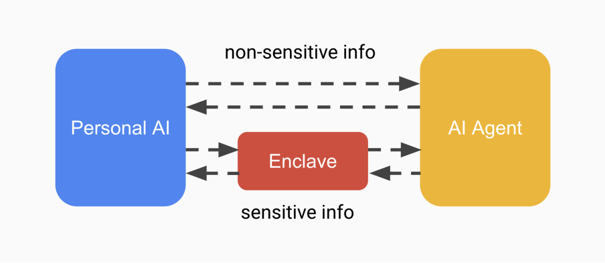

Figure 3: Agent Communication Protocol

By segregating non-sensitive and sensitive information in this manner, the user benefits from personalized AI assistance while maintaining the privacy and security of their personal data. The last step is verification; after the interaction is concluded, the Personal AI generates a proof of the outcome which is passed to the PIN verifier onchain.

References

[1] PIN AI: The Open Platform for Personal AI. https://www.pinai.io/post/pin-ai-the-open-platform-for-personal-ai

[2] Unichain Whitepaper. https://docs.unichain.org/whitepaper.pdf

[3] Introducing Rollup Boost. https://writings.flashbots.net/introducing-rollup-boost

[4] Chris Dixon. *Read Write Own: Building the Next Era of the Internet*. 2024.

[5] John Battelle. *The Search: How Google and Its Rivals Rewrote the Rules of Business and Transformed Our Culture*. Portfolio, 2005.

[6] POAA Whitepaper. *Proof of Active Authorship (POAA)*. Available at: `https://staking.olas.network/poaa-whitepaper.pdf`, 2024.

[7] Zhang, Fan, et al. *Town crier: An authenticated data feed for smart contracts*. 2016.

[8] Zhang, Fan, et al. *Deco: Liberating web data using decentralized oracles for tls*. 2020.

[9] Luo, Zhongtang, et al. *Proxying is Enough: Security of Proxying in TLS Oracles and AEAD Context Unforgeability*. 2024.