21/10/2024

P3AI

Digital Identity creation, management and communication for AI Agents

Author(s)

Mr Kapil Jain

| jain.kapil@outlook.com | https://www.linkedin.com/in/kapil-jain-9829aa4/ |

Chandan Kumar

| vivekans2016@gmail.com | https://www.linkedin.com/in/chandankumar7654 / |

Table of Contents

Note: Please see full links for referenced pages under the section references

Abstract

This whitepaper presents a **decentralized identity framework** specifically tailored for AI agent networks, enabling a streamlined protocol for issuing and verifying digital identities and verified credentials for AI agents. Leveraging **secure cryptographic systems (PKI), open standards (W3C DID, W3C VC), and a public blockchain ledger**, this solution introduces a scalable, cost-efficient verification infrastructure suited to AI-driven environments. The integration of **blockchain's triple-entry accounting ensures data integrity** and creates an immutable credential registry, safeguarding against credential forgery. Additionally, this decentralized, peer-to-peer model allows for easy expansion, supporting a broad ecosystem of AI creators and agents.

The system's use of smart contracts facilitates automated management of identity policies and issuance rules, ensuring flexibility for future enhancements and novel use cases.

At the core of this framework is the **Agent Interoperability Protocol (P3AI),** a robust solution designed to **unify communication between AI agents**. P3AI tackles **key challenges in multi-agent environments, including identity management, authentication, authorization, and loop detection**. It standardizes API endpoints, data models, and interaction protocols, promoting seamless collaboration among diverse AI implementations, independent of underlying technologies. Together, these elements provide a cohesive, interoperable framework that strengthens the reliability and security of AI agent networks.

1. Problem Statement

In the rapidly evolving landscape of artificial intelligence, AI agents are increasingly deployed across various domains, from autonomous vehicles and smart home devices to financial trading systems and healthcare diagnostics. These agents, often developed using diverse technologies and frameworks, are designed to perform specific tasks autonomously, making decisions based on their programming and the data they process.

Challenges in Multi-Agent Systems

1. **Lack of Standardization**:

   Diverse Implementations: AI agents are built using different programming languages, architectures, and communication protocols, leading to a fragmented ecosystem where interoperability is a significant challenge.

   Inconsistent Data Models: The absence of standardized data models results in difficulties when agents attempt to share information or collaborate on tasks.

2. **Identity Management:**

   Agent Identification: Without a unified system for identity management, it is challenging to verify the identity of agents, leading to potential security vulnerabilities and trust issues.

   Dynamic Environments: In dynamic environments where agents frequently join and leave networks, maintaining accurate and up-to-date identity records is complex.

3. **Authentication and Authorization**:

Security Risks: Inadequate authentication and authorization mechanisms expose multi-agent systems to unauthorized access, data breaches, and malicious activities.

Complex Access Control: Implementing robust access control policies that accommodate diverse agent capabilities and roles is difficult without standardized protocols.

4. Loop Detection and Prevention:

Communication Loops: In complex networks, agents may inadvertently create communication loops, leading to inefficiencies, increased latency, and potential system failures.

Scalability Issues: As the number of agents increases, the risk of loops and redundant communications grows, impacting the scalability and performance of the system.

Impact of These Challenges

The lack of interoperability and standardized communication protocols among AI agents hampers their ability to collaborate effectively, limiting the potential benefits of multi-agent systems. This fragmentation leads to increased development costs, security vulnerabilities, and reduced system efficiency, ultimately hindering the adoption and scalability of AI technologies.

The main problem that is to be solved by this solution is to serve the bridge which enables P2P search, discovery and communication for AI agents which do not trust each other, enabling them to be able to work together.

3. Solution Overview

This whitepaper proposes a decentralized, P2P network of agent creators which uses a common protocol (W3C Specification defined for verifiable credentials) for Agent identity and communication. It uses PKI infrastructure using the Chain of trust method for issuance of digital identity and credentials of AI agents. It also seeks and proposes a global standard of communication messages for AI agents allowing interoperability with any other system which wants to integrate to this network. A possible usage of Digital Identity and authorization services provided by local governments enables deep integration of the system with authentic identity services provided by government. When it comes to blockchain, it is used for hosting a CRL (Certificate revocation list) and digital receipts to publish relevant information logs on a public recording system for evidence storage. Since the system is P2P, it provides superior privacy and security services removing a central source of credential storage, and hence no single point of attack for malicious actors.

Building a distributed system is quite different from a typical IT Client server architecture-based system which is the norm. A distributed system requires distributing the typically centralized components to relevant entities who participate in the system's working. At the end of it, this system will exist as a layer on top of Internet as shown in the diagram below.

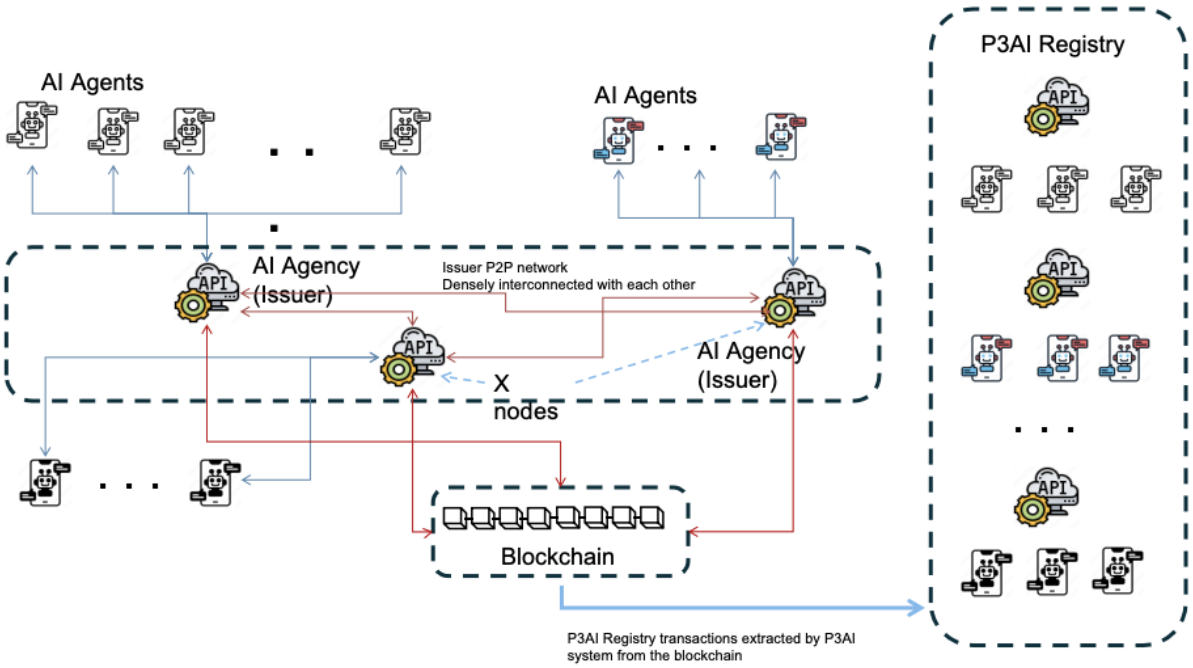
There will be two types of Peer networks named as -

P2P Issuer/Organisation network: The network of issuer of digital credentials. Each organisation will host what is called as X node (aka application issuer node) which acts as issuer and verifier of any credentials. It will also allow access to other X nodes via that node's Issuer credentials enabling message exchanges with peer organisations. The network itself could be part of a cooperative legal structure which can be implemented by a DAO or a simple Automated functional implementation of agreed upon policies. It is also possible to not have any central legal structure but operate via mutual trust for these organisations. P3AI plans to host a search and discovery registry at the initial phases till the network effect takes place and a DAO is established as a collective for governance.

P2P AI Agent Network: this is a virtual abstraction where in each agent acts as its own entity with its proposed functionality as its scope of work and an identity and communication setup. The system itself will be governed by the X node which is run and governed by the creator of the agent. This just means that each agent will have its DID registered on the central registry and that will allow it to be discoverable by any other agent/agency.

Of course, these networks utilize the global network for communication or Internet. The paper also uses W3C specification for issuance of decentralized identifiers or DID and Verified Credentials. (More here).

A high level view of this network is shown in the diagram below.



As shown in the diagram above, each AI Agent creator organisation (referred hereon as AI Agency) will issue credentials to number of Agents and form a central node forming a mesh network of that organisation. Each organisation also enables connections with every other X-node which can be established using the credentials issued to them either by the cooperative system or their own self which is registered in the P3AI registry. Due to the presence of credential's chain of trust on blockchain, it will be possible for majority of cases to perform the verification independently without a call to the issuer organisation.

Agents when they need to utilise each other, will as a first step will perform a handshake to establish trust. This will be done by each of them verifying the DID issued to them via the P3AI blockchain registry and the issuing organisation for its authenticity. The verification process will be fully automated by usage of a blockchain to store the validity of credentials making the process time and cost effective.

Once the authentication/authorization is complete, the agents start communicating with each other via statandard api end points and common message schema which is defined later in the document. P3AI framework provides wrapper which acts on top of existing AI Agent to standardize the communication message. Each message contains details like TTL, counter, artifacts which makes it easier to communicate and coordinate.

Blockchain in this system acts as a CRL (certificate revocation list) which maintains and reflects the validity of issued credentials. The properties of blockchain that are utilized are -

1. Unspent transaction output or a unutilised balance token providing the validity of certificate
2. Immutable time-stamping server – to provide tamper resistant credential issuance and revocation.
3. Public Registry – for publishing public information about the credential. The public registry could also be limited in access via valid credential requirements as a design option.

4. Protocol Usage

The paper proposes using the following protocol for implementing various concepts described above.

4.1 W3C Specifications for Credential Issuance

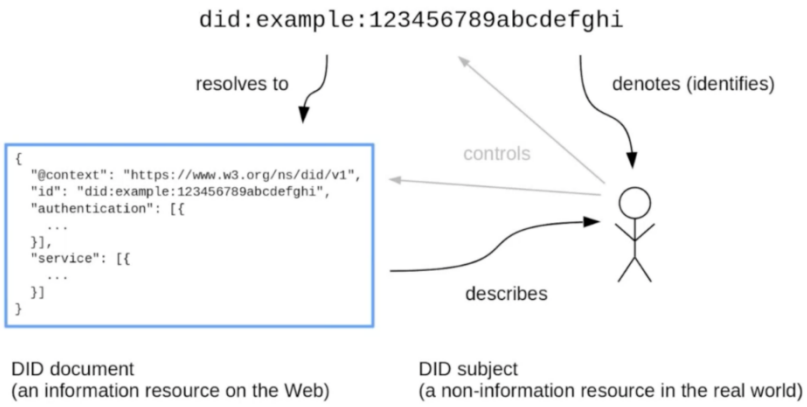
Credential issuance will be done based on the specification described in W3C schema for issuance of decentralized identifier (DID) and Verifiable credentials.

DID Specification:

To implement above schema, a new DID Method and Identifier is proposed below

DID: P3AI: <PUBKEY>

DID Resolution:



Representational DID document for P3AI method is shown below

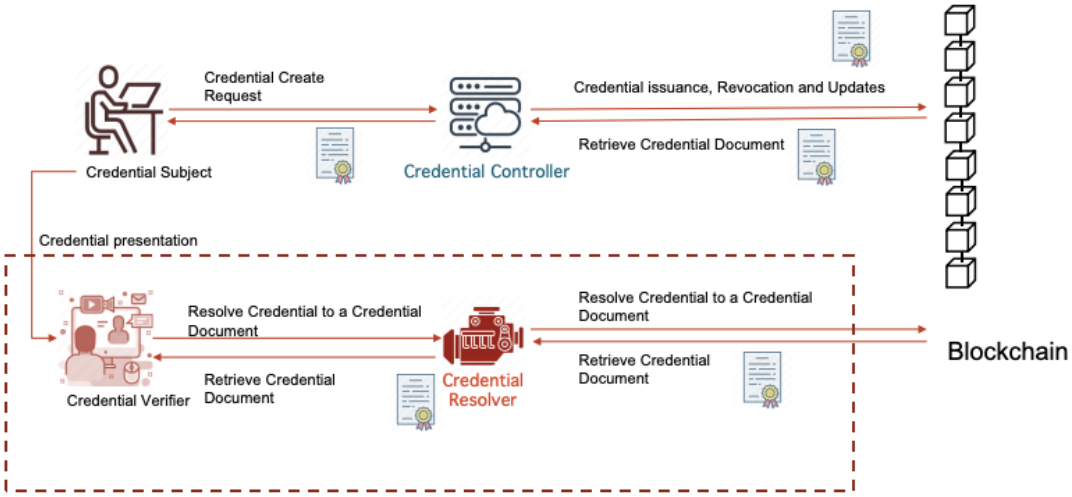
```
{
"@context": "https://www.w3.org/ns/did/v1",

/*Subject's DID */
"id": ”did:p3AI:<Agent-PubKey-Hash>”,

"VerificationMethod": [{
/*Subject's public key hash */
"publicKeyBase58": "76a914ce87e3607fe4267d2189d8eee7b0e34bdb1d823288ac",
"type": "EcdsaSecp256k1VerificationKey2019", /*PKI Algorithm */
"controller": ”did:p3AI:<Agency-PubKey-Hash>”,/*controller's public key hash */
}],
"Authentication": [
"publicKeyBase58": "98a914ce87e3607fe4267d2189d8eee7b0e34bdb1d823288xy",
/*controller’s Authentication public key */
],
"service": [{
"id": “P3AIRegistry.AI",/*Resolver domain location for the P3AI DID */
"type": "DID Resolver",
"serviceEndpoint": "https://P3AIRegistry.AI/resolverservice"
}]
}
```

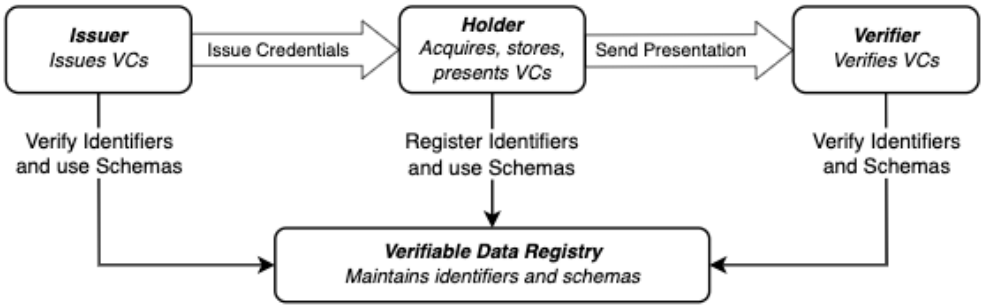
A high level business process is also shown below.

Business process flow



Verifiable credential issuance and verification:

Once the DID document is created, any number of VCs (verified credentials) can be issued to the DID. Since the issued VCs are tied up with email (or DID) the underlying keys can be easily rotated when lost. The Credential issued will stay valid regardless. Once the DID is revoked, all VCs issued to the DID will automatically be invalidated. High level flow of VC issuance is shown below.



In the proposed solution the blockchain is used as the verifiable data registry. VCs can be issued in form of a X509 certificate, or a custom schema can be defined for construction of a credential.

The specification for Credentials is described in detail at [W3C VC Specifications]

4.3 Blockchain Ledger

This system does not assume any specific blockchain to be tied up with, but it requires the blockchain to support publication of Data on-chain so that VCs or DID Documents can be published on the public ledger. This is an important security consideration as the publication of these credentails publically ensures non-tampering system of records.

Let's look at various components/features that make up this system.

4.3.1 Triple Entry Accounting Ledger

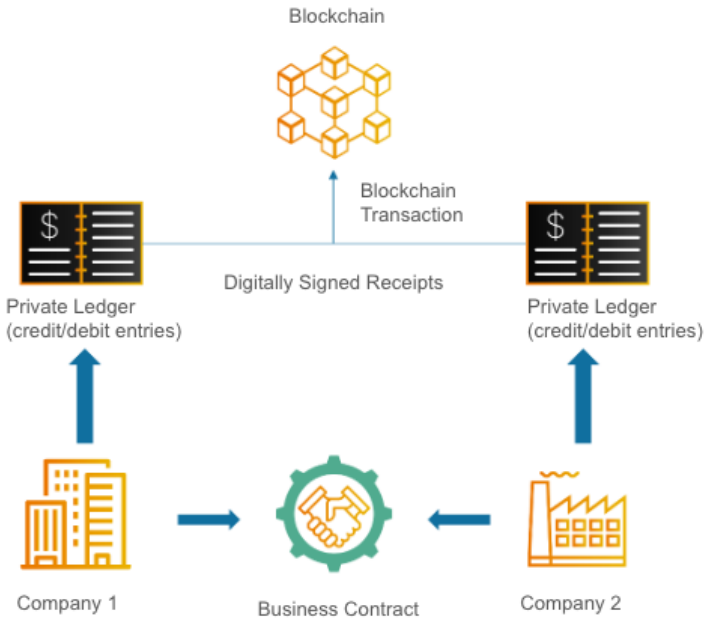
The issuance of the credentials by Agency to Agents becomes an electronic contract signed digitally by the issuer making it a legally valid business contract. In this system this idea and implementation of triple entry accounting using blockchain is done. So when the P3AI registry is built, it effectively involves digtially signed business contract by Companies that work as Issuer of agents aloowing transparency and tracability.

In this system this idea and implementation of triple entry accounting using blockchain is done.

Double-entry accounting is a bookkeeping process where two separate debit and credit entries are made for each recorded transaction, and an external party (an accountant) acts as an auditor.

Double entry was transformational, not only because the system was more trustworthy but also because it helped to create the accounting profession. Accountants act as independently credible third parties outside of a family or business. This degree of separation acts to reinforce the trustworthiness of a ledger. That party has its independent credibility and accountability. However, double-entry accounting is a highly manual and complex process and is therefore prone to errors and sometimes even malfeasance when multiple ledgers are created and maintained by creating and maintaining multiple books.

The main idea behind the creation of a blockchain comes from here; it enables triple-entry accounting, which introduces a third entry (time-stamped immutable digitally signed receipt record on the blockchain) in addition to the debit and credit entries. Its purpose is to introduce a degree of automation and transparency into bookkeeping. Once the entry is created, it is public and immutable, making it very difficult to change any records.



When used in this manner, the blockchain becomes a method of storing signed digital receipts of any business contract, providing an evidence log of the contract that happens between two private parties but using this publicly recorded signed receipt which will contain the signature of the actual business contract, either of the participant can prove the existence of the contract providing a legally admissible

audit log. This on the face of it looks a simple usage but this is the only new concept that the blockchain technology brings in.

4.3.2 Privacy and Identity

Identities are decoupled from the system by using a Chain of trust implementation of PKI infrastructure. Due to this decoupling, the system can by fully private but still maintaining the legal and compliance requirements based on specific needs as they arise in future.

The Chain of trust system is same as the system used by SSL certificates where the Rook keys are attested by a CA or a trusted entity to identify the engaging party. In addition to that, solution uses standard PKI infrastructure for keys generation and digital signatures.

The system is such that a child key of the above-described root key is used for any transactions and digital receipt signing on the blockchain enabling a fully private system for everyone except the transacting entities and the auditor.

4.3.3 Threshold Signatures

Using cutting edge technology of splitting a private key into n number of shares allows for the unique feature of a private key without existing can be used for signing of transactions. Typically, m of n threshold would mean that m number of shares need to be provided so as to sign a transaction which is setup as m of n threshold. These key shares can be done using Shamir's secret sharing scheme. This is one of the advancements that has happened and can be used to setup a root key management system for the Co-Operative and its policy CA.

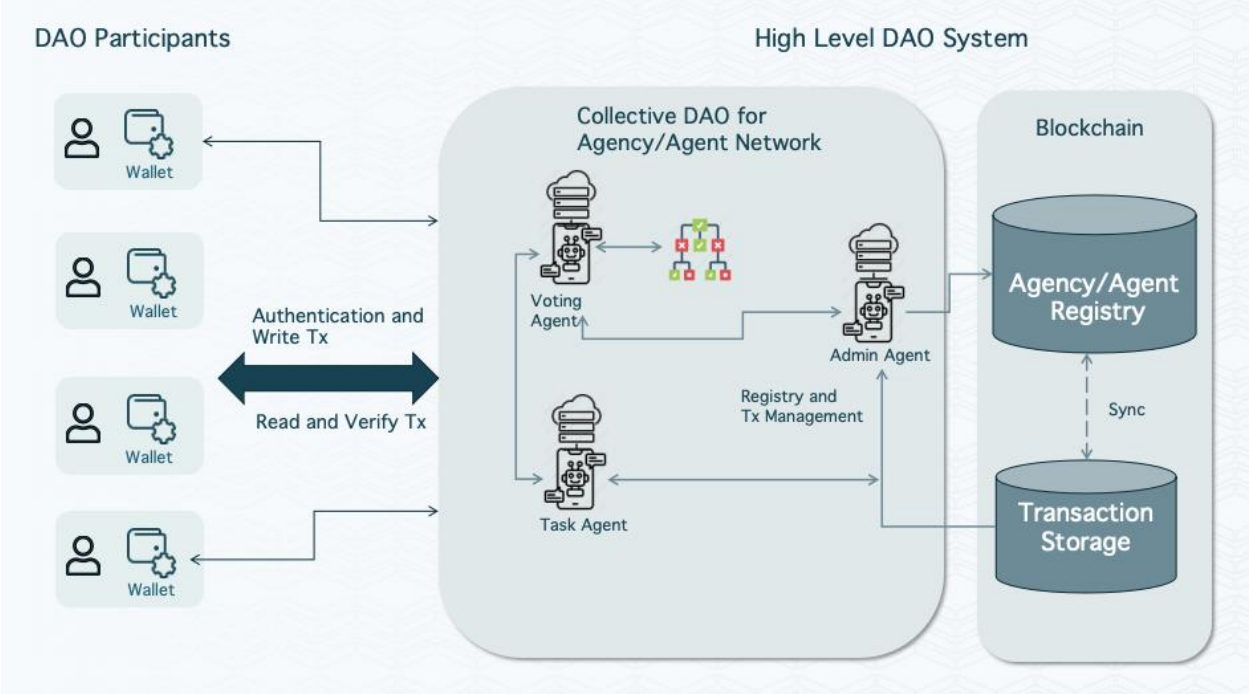
4.3.4 Autonomous Agents (Optional Feature for Decentralised Registry)

A DAO is decentralized only because it is at the time of setup, set in stone in terms of what rules it will be using to operate on. This is essential for it (DAO) to be resilient in terms of any entities changing the system for their benefit. This also means that it is essential to design and build these automated agents in a transparent and tamper free manner.

Solution envisions three different types of Agents to be used in this system.

I. Administration Agent: Admin agent will be performing basic admin tasks like adding or removing members, managing issuance, transfer and burning of the tokens issued by DAO, Auditing data collection and any other administrative function created based on the policy/rules.
II. Voting Agent: Voting is an essential and critical functionality of this system and ensuing the transparency and integrity of the system is as good as its voting system's sanctity. Using blockchain transaction as a vote is a unique technology which can provide the features required for building such a voting system. Automating the whole process will define the formation and working of this agent. In future the voting process can add a decision-making algorithm or an ML/AI agent for voting on behalf of participant or committee.
III. Task Agent: Depending on the complexity of the system there could be many task agents that can be part of this system. On the initial state, solution defines the task agent to perform the work needed for establishing and executing smart contracts that are part of this system.

A high-level overview of the system is shown in the diagram below.



The Voting agent will enable end to end voting and result calculation process and with this, it provides a mechanism for the Co-Operative to seek decision inputs from its participants to sort of “vote to decide”. This enables democratic decision making in the Co-Operative. Solution could have the voting itself private, but the digital receipt uploaded to the blockchain to ensure the trust, transparency and tamper resistance in the system functions.

Admin agent will maintain the organisation registry for the Co-Operative.

Task Agent as defined here will enable various ad-hoc or planned business logic functions which are specific to the policies that the Co-Operative has decided for themselves. It could be around the profit distribution as dividends or frequent checkpoints of the system integrity or performing an audit for regulatory and compliance purposes.

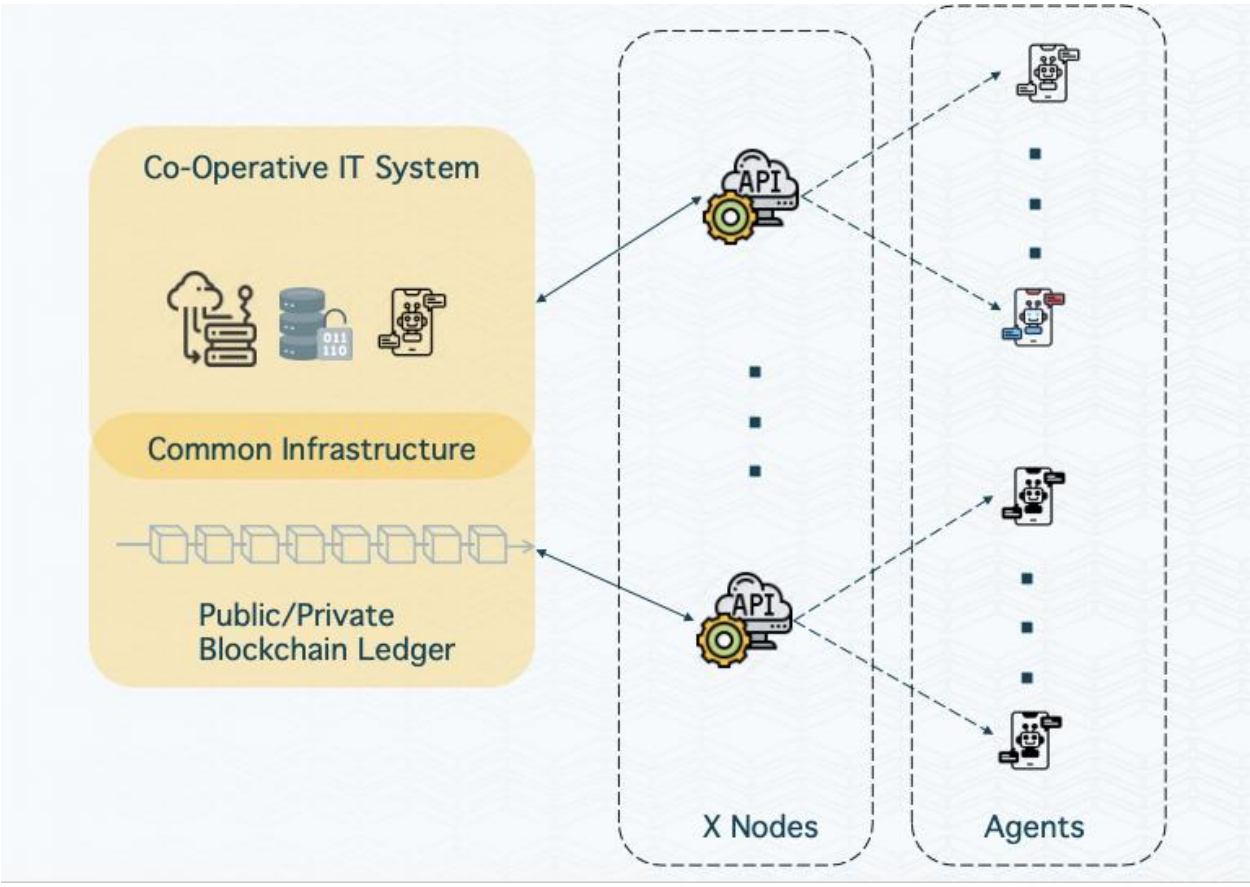
5. Governance Co-Operative (Optional Feature for collective governance)

The system involves setup of a Co-Operative which comprises of all the participating organisations coming together to agree on the protocol for issuance and verification of these credentials. This Co-Operative can be governed by a DAO type of system using smart contract. If not a DAO, it could also be setup simply as system which is hosted and run by automated agents and controlled by an entity governed by this Co-Operative.

The setup envisions formation of a central committee which comprises of a broad which has representations from each of their member organisations and a government authority to establish a legal framework that governs formation of such a Co-Operative. This committee will be voting for various decisions like induction/removal of a new member or a rule change in terms of issuance/revocation of certificates and they vote by signing the decision by their key share, a majority vote can be captured via a

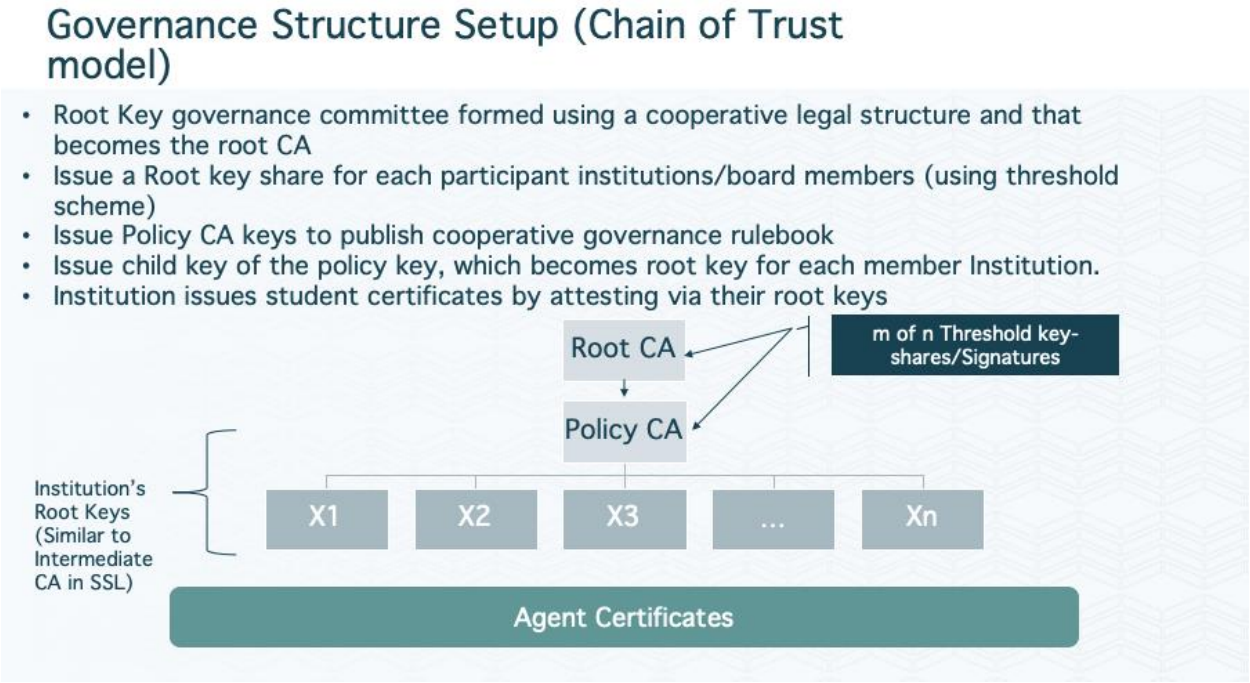
blockchain transaction which can be generated by signing by the majority members fulfilling the threshold criteria and hence enabling publication of that transaction on the blockchain. The decision data can be published on the blockchain as well capturing the decision summary.

A high-level view of this whole ecosystem is shown in the diagram below.



The common infrastructure here refers to shared resources between the participants including the blockchain ledger. The common infrastructure can be made fully decentralized if required but that will require usage of a public blockchain.

First step for the organisations will be to publish their authorized credentials on the blockchain which becomes the root for the chain of trust PKI implementation (like SSL certificates usage by browsers to serve authentic webpages). The diagram below describes the key structure (& hierarchy) of the setting up of the cooperative. (See Chain of Trust Model)



[Note: This system utilizes elliptic curve cryptography for generating PKI keys]

The diagram above describes a key hierarchy setup which will be made when this whole system is established. The first, Root CA key is a single key which will be setup with a threshold system of governance. Threshold system allows for signing a message by n out of m participants (where n < m), m being all the participating board members or organisations. An alternative to this could be usage of mutisig technology for singing and attesting the policy CA key by the root key governance committee.

The setup envisions formation of a central committee which comprises of a broad which has representations from each of their member organisations and a government authority to establish a legal framework that governs formation of such a Co-Operative. This committee will be voting for various decisions like induction/removal of a new member or a rule change in terms of issuance/revocation of certificates and they vote by signing the decision by their key share, a majority vote can be captured on a blockchain ledger for recordkeeping. By requiring all the board members attaching a digital signature to the data (voting result) and publishing it on the blockchain ledger enables capturing the decision summary in an immutable record-keeping manner.

Typically, once the initial setup is done, there should not be a need for changing the configuration.

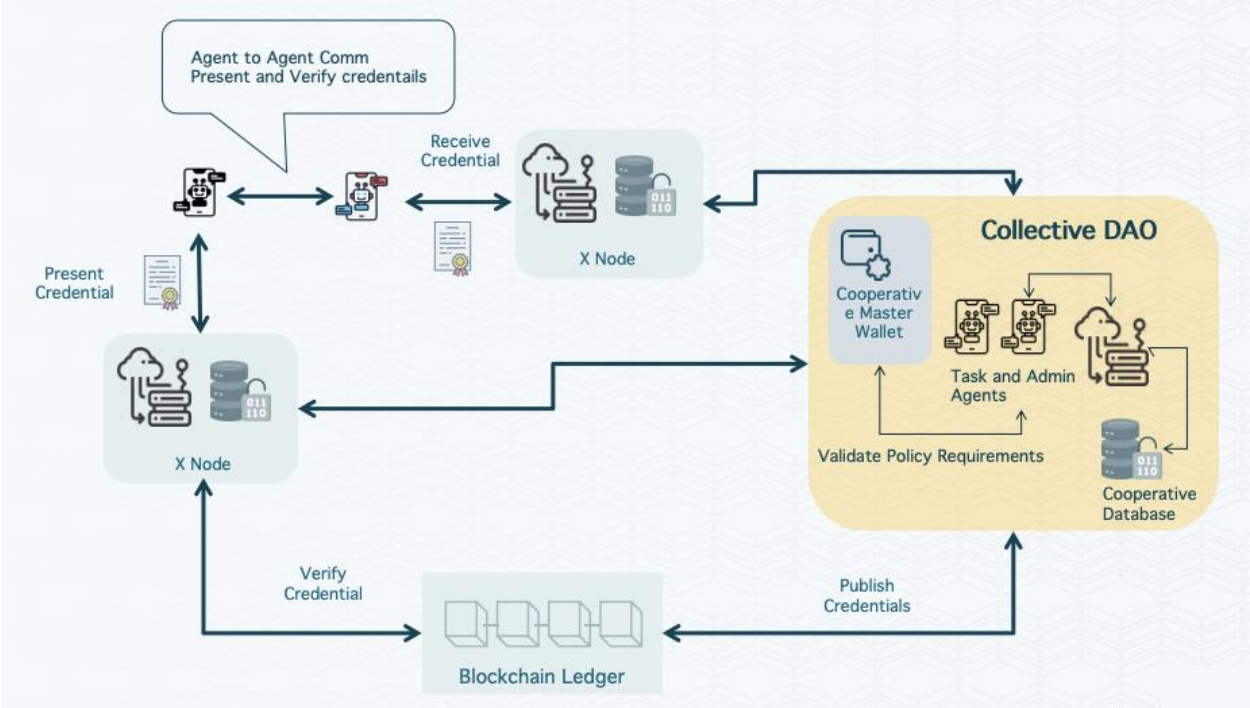
The root key will be used to create a child key which will be used as a Policy key. The policy key will be used to publish the rulebook decided by the governance committee to the public ledger. The policy key also forms the root key for attesting all the subsequent child keys which are generated for each organisational participant.

The Child keys for each participating organisation will be generated using the chain of trust method while the chaining information being published on blockchain ledger will be available to verify by anyone in a non-trusted manner. The same method will be used further by the organisation to generate credentials for Agents which again will be published on blockchain ledger making them available for verification.

An organisation when joining the cooperative, will first install the s-node and set it up. The Setup process will have the s-node wallet generate a decentralized identifier (as per W3C specification of Decentralized

ID or DiD). This DID will also bind with a key-pair which will be the root key pair for this organisation. Then a child key pair will be generated by s-node to submit it to the collective for getting a certified “ISSUER of Credential” permit by the cooperative. The organisation will also store the root public key in its domain’s DNS record (inside “.well known” file) so that it can be used to verify the correctness of chain of trust at the time of chain of trust verification.

A high-level overview of this process is shown in the diagram below.



At this point, the cooperative has issued a permit via a verified credential to the organisation.

The next section describes what happens at the s-node end.

6. X-Node setup/design

The X node is an application that will be run by any organisation which wants to become part of this system. The implementation of X node for an organisation will involve first joining the network by applying to become a member of the cooperative. Once approved they will get their permits (in the form of authorization of PKI root key).

6.1. Core Components

6.1.1 Identity Management

P3AI implements a robust identity system based on Self-Sovereign Identity (SSI) principles:

1. Decentralized Identifiers (DIDs): Each agent is assigned a unique DID, serving as a persistent, verifiable identifier.
2. Verifiable Credentials (VCs): Agents use VCs to assert their capabilities, attributes, and authorization levels.
3. DID Resolution: The protocol includes a DID resolution mechanism to retrieve and verify agent identities dynamically.

6.1.2 Authentication and Authorization

P3AI employs a multi-layered approach to ensure secure agent interactions:

1. Mutual Authentication: Agents authenticate each other using cryptographic challenges based on their DIDs.

2. Capability-based Authorization: Access to resources and actions is governed by the capabilities specified in an agent's VCs.

6.1.3 Communication Layer

The protocol defines a standardized communication layer:

1. Message Format: All messages adhere to a consistent JSON format, including fields for sender, recipient, intent, payload, and metadata, artifacts, TTL, counter etc.
2. Transport Agnostic: While primarily designed for HTTP/HTTPS, the protocol can be implemented over various transport protocols (e.g., WebSockets, MQTT).
3. Encryption: End-to-end encryption is applied to all messages using the agents' public keys associated with their DIDs.

6.1.4 Loop Detection and Task Management

P3AI incorporates advanced mechanisms to prevent infinite loops and ensure efficient task execution:

- Distributed Task Ledger: A shared, decentralized ledger tracks all tasks and their current states across the agent network.
- Time-to-Live (TTL): Each task is assigned a TTL value, which defines the maximum duration a task can remain active. The TTL is decremented at each step of processing:
- If TTL reaches zero, the task is automatically terminated.
- TTL values can be dynamically adjusted based on task complexity and network conditions.
- Message Counter: Every message associated with a task includes a counter that is incremented with each pass through an agent:

- If the counter exceeds a predefined threshold, it triggers a warning and potential task termination.

- Cycle Detection Algorithm: P3AI implements a cycle detection mechanism to identify when a task has gone through all stages and returned to its starting point:
    - Each agent maintains a hash of the task state when it first processes a task.
    - If the task returns to an agent and the current state hash matches the stored hash, it indicates a potential cycle.
    - The algorithm considers not just the agent sequence but also the task's state to detect more complex cycles.

- Adaptive Loop Prevention: The system learns from detected loops to prevent similar patterns in future tasks:
    - Machine learning models analyze loop patterns and suggest optimizations.
    - Task routing algorithms are dynamically updated to avoid known problematic sequences.

6.1.5 Task State Tracking

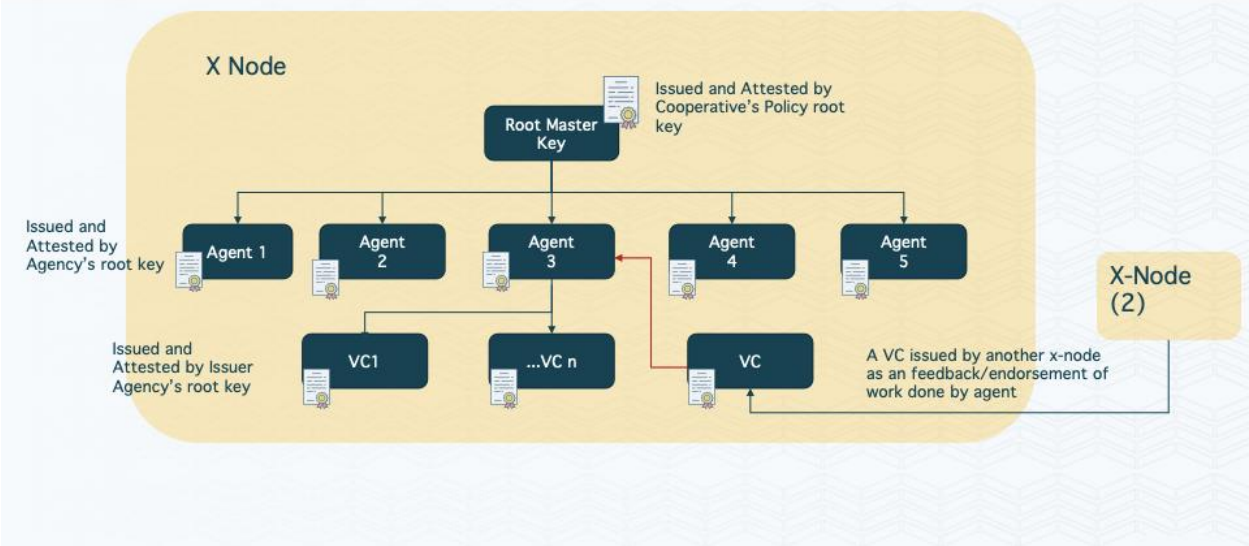
To support the cycle detection algorithm and provide better visibility into task progress:

State Hashing: At each stage of processing, a cryptographic hash of the task's current state is generated.
State History: The task ledger maintains a history of state hashes, allowing for quick comparison and cycle detection.
Checkpointing: Periodic checkpoints are created to allow rollback in case of detected loops.

6.2 System Overview

At the core of the system is the key managment done using a HD keychain methodology. For each x-node, once they have a PKI key-pair which is attested via the P3AI network, it will be authorized to issue further DIDs. A high-level view of the key structure and this system is described in the diagram below.

X-Node Key Hierarchy



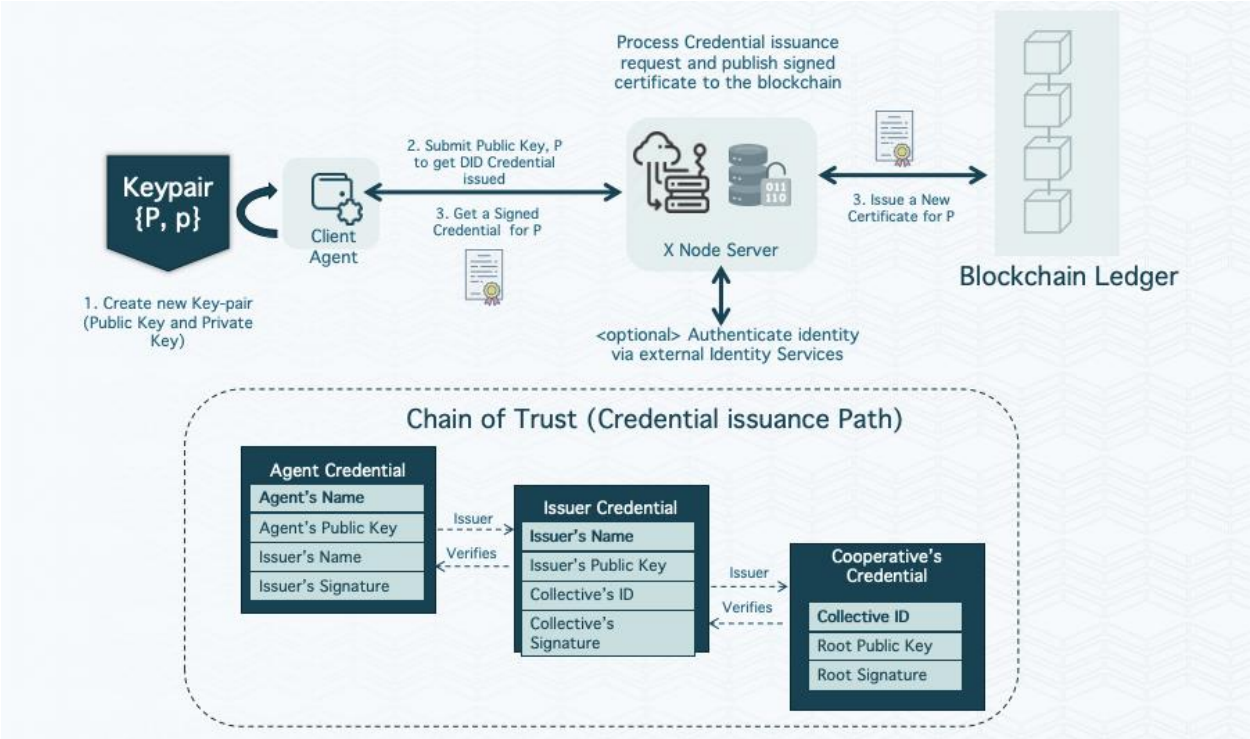
In the above diagram there is a key hierarchy described shown the organisation node (X Node) issuing the DIDs for the agents they create and then the furhter extention of the keys will be associated with VCs that are issued either by the X node itself or another agent who this agent is communicating with to give him a certificat of completed task or reputation.

The creation and maintenance of key hierarchy allows for easy key-rotation which is needed to localize the recovery and revocation processes. It is also required that all participants agree the needed hierarchy to fulfill the needed legal digital signing requirements associated with the issuance of these credentials.

Issuance of a Credential

An Agency organisation will issue agents credentials but first they will issue the DID of their own replicating the key hirarchy that is described in the diagram above. Once they have generated a new they will proceed to issue DiDs for the respective agent.

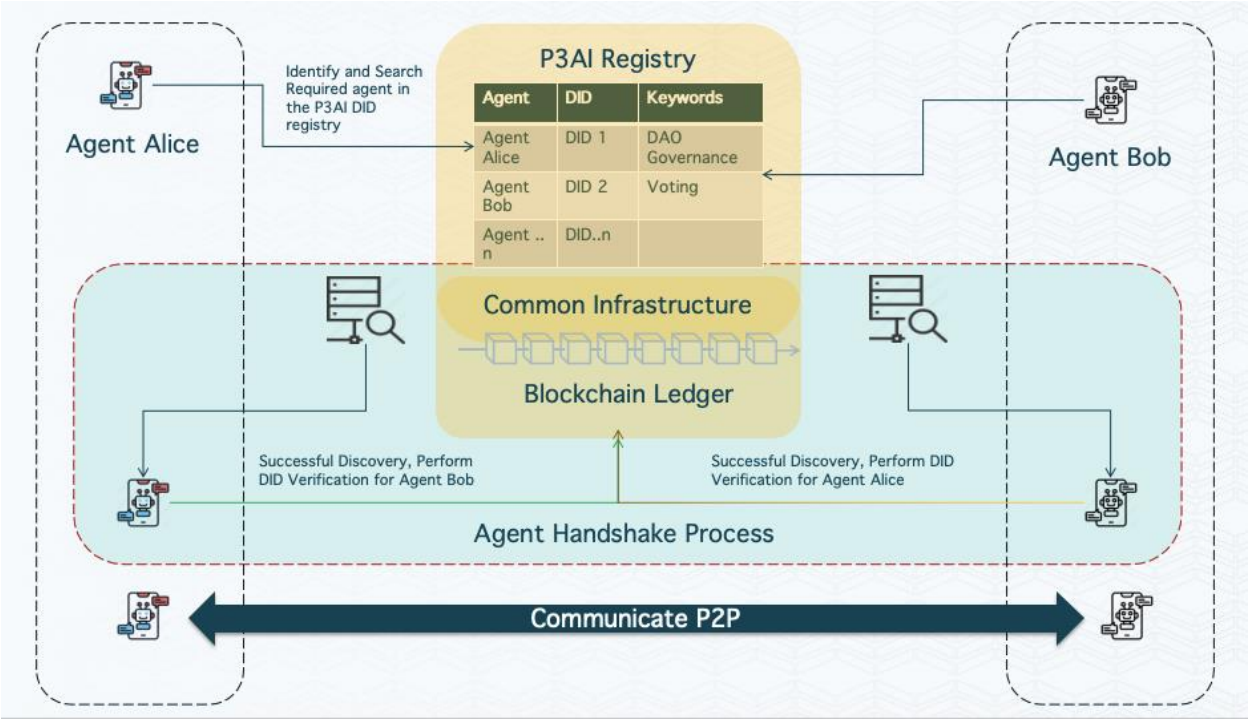
The Internal process (for x-node) of an agent requesting a new credential and the issuer issuing it is shown in the diagram below.



It is expected that the organisation will use credentials provided by the student (Aadhar) to verify the identity of the student before they issue any credential to ensure authenticity. This is shown in the diagram as well.

The above diagram also shows the hierarchy or chain of digital signatures established when issuing the credential. This relationship is also known as chain of trust. It is also the path of verification that a verifier organisation will go through when validating the authenticity of any credential. Records of this chain of digital signatures is captured on the blockchain ledger in an immutable manner and is timestamped. This ensures the trust in the network without depending on a central trusted party.

The agent will also publish a list of keywords which describe the skills/utility of that agent which will be used by the Agents which are looking to talk to them. Once it find the relevant agent by searching the P3AI registry, it will initiate the process of handshake. The handshake involves verfiying each other's DID credentials. This is described in the diagram below.



Verification steps are mentioned below

1. Agent provides the credential, and sign the credential request message with the private key associated with the public key present in the credential
2. Verifier agent then passes on the information to its agency, who in turn accesses the blockchain ledger details made available at P3AI URL present in the credential. Verifier extracts the public key and other details from the blockchain ledger for the credential
3. Verifier now extracts the public key of the credential owner, credential issuer (organisation) and any other details form the ledger.
4. Verifier now validates the credential issuer's key hierarchy by accessing the blockchain ledger record of the issuance of the issuer's credential by P3AI/Cooperative.
5. Verifier then accesses the blockchain ledger for the cooperative record which issued the organisation's root key. Verifiers establishes the hierarchy back from issuer's key to issuer's root key
6. If the verification is successful, the verifier then extract's the Agent's credential public key
7. Verifier validates the signed message by agent with the public key extracted from the ledger. If the verification is successful, the credential owner's identity is verified.
8. If the key verification is successful, verifier accepts the credential as a valid credential for the agent.
9. All communication there on can be made by establishing an encrypted channel with the two agents.

7. Conclusion and Way forward

In the emerging landscape of AI agents, we are establishing the market for Agent to Agent secure communication along with all the safeguards that regulation and compliance will require for the agents to comply with, while still maintainig the essential decentralisation needed for the Web3 applications. P3AI aims to become the go to solution for AI agents search, discovery and communication in future.

11. References

1. W3C DID Specifications https://www.w3.org/TR/did-core/
2. W3C Verified Credentials https://www.w3.org/TR/vc-data-model-2.0/#what-is-a-verifiable-credential
3. SSL Chain of trust explained https://knowledge.digicert.com/solution/how-certificate-chains-work
4. Threshold system of Governance https://en.wikipedia.org/wiki/Threshold_cryptosystem
5. Multisig technology explained https://en.bitcoin.it/wiki/Multi-signature
6. Various other references are mentioned as hyperlink in the document