network working group
Internet-Draft
Intended status: Informational
Expires: 23 April 2026

L. Zhang
B. Liu
Huawei Technologies
N. Geng
Huawei
Z. Wang
Independent

20 October 2025

Agents Networking Architecture for Enterprise and Broadband
draft-zl-agents-networking-architecture-00

Abstract

This document introduces agents networking architecture and defines the core components of agent networking, as well as typical interactions among these key components.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 23 April 2026.

Copyright Notice

Table of Contents

1. Introduction

With the rapid development of artificial intelligence, single-agent systems have gradually revealed their limitations in handling complex, multi-task, and cross-domain scenarios. Agent networking, as a core paradigm for breaking through individual capabilities and achieving collective intelligence, has become a key trend in the future development of agents.

This document introduces agents networking architecture and defines the core components of agent networking, as well as typical interactions among these key components.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. IoA architecture and Components

2.1. Overview of Agents Networking

The overall architecture of agents networking is shown in Figure 1.

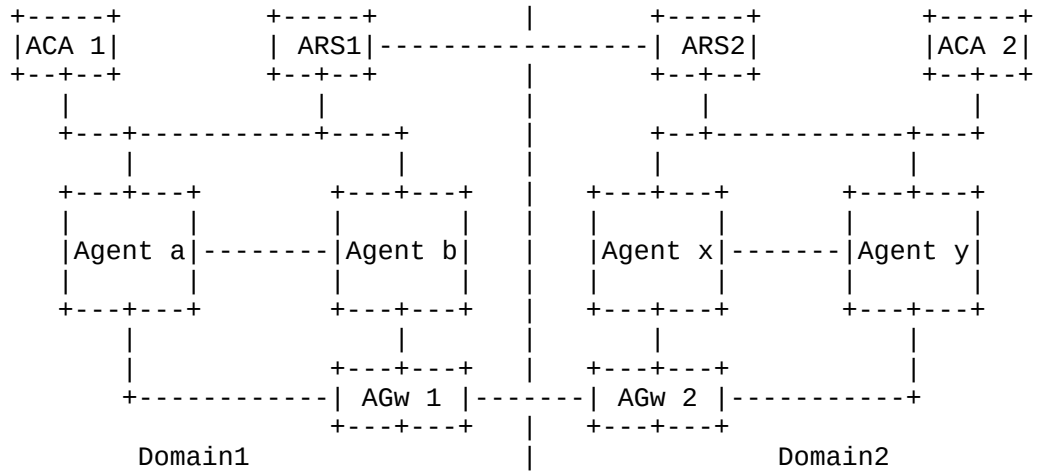

Figure 1: Agents Networking architecture

The architecture of agents networking consists of four key components, the Agent Certificate Authority(ACA), the Agent Registration Server(ARS), the Agents, and the Agent Gateway(AGw).

2.2. Agent Certificate Authority

An Agent Certificate Authority (ACA) is essentially a trusted third-party organization that issues and manages digital certificates for intelligent agents (Agents). Its core function is to ensure the authenticity of Agent identities and the security of their interactions. By issuing digital certificates that contain key data such as Agent identity information and public keys, it enables other Agents or systems to verify the identity of the Agent, preventing the access of counterfeit or malicious Agents.

The major functions of ACA include Certificate Issuance and Certificate Management.

* Certificate Issuance: It verifies the real identity of an Agent (including developer information, affiliated organization, functional permissions, etc.). After verification, a unique digital certificate is generated and issued.

* Certificate Management: It is responsible for the full-lifecycle management of certificates, including certificate renewal (renewal before the expiration of validity period) and revocation (when an Agent is deregistered or its key is leaked).

2.3. Agent Registration Server

An Agent Registration Server(ARS) is specifically responsible for Agent identity registration, capability profiling, and accurate discovery-matching. Acting as the "identity management center" and "resource directory library" in the Agent ecosystem. The major functions of ARS include agent registration, agent information management, and agent discovery. The ARS can be either centralized or distributed, but regardless of the form, information synchronization between different servers is essential. For easily deployment, the ARS also can be integrities in the AGw.

* Agent registration: The agent sends a registration request with its information, upon the verification is passed, then the ARS generates a unique "resource ID" for the Agent, creates a profile in the "Agent Database" to record all the agent information.

* Information management: It is responsible for the full-lifecycle management of Agents profiles, including update (when a gent’s information changes) and delete (when an Agent is discarded).

* Agent Discovery: The ARS maintains a dynamically updated "Agent Database" that records key information of all Agents registered at this ARS and synchronized from other ARSs. An agent needs collaboration just need to send a capability request to the ARS, then the ARS will quickly filter out matching target Agents and return them information to the requested agent.

2.4. AI Agents

The functions that should be supported for AI Agent interconnection include task orchestration, task management, and session management.

* Task orchestration: It breaks down complex tasks into subtasks, assigns them to Agents with corresponding capabilities, and clarifies the collaboration process. Based on the capabilities of each Agent (e.g., a Flight Agent specializes in ticket booking, a Visa Agent specializes in visa processing), it plans the execution sequence and dependency relationships of tasks to ensure efficient resource utilization.

* Task Management: It tracks subtask progress, handles exceptions, and ensures the overall task progresses as planned. It monitors the execution status of each subtask (e.g., "pending," "in progress," "completed," "failed"), records task output results (such as flight confirmation numbers and visa documents), and addresses issues during execution.

* Session Management: It maintains the continuity and integrity of interactive sessions, ensuring historical information is traceable and context is not disconnected. It manages the lifecycle of conversations between Agents, records all interactive content (e.g., instructions, feedback, problem discussions), and provides contextual support for subsequent interactions to avoid repeated communication.

2.5. Agent Gateway

An AGw essentially serves as a "Connection Hub" and "Security Steward" in the process of agent communication. Its core role is to resolve interconnection issues between different Agents of different domains while ensuring secure, efficient, and orderly communication. The AGw should support the following functions:

* Route Management: When it comes to cross-domain agent communication, an agent or AGw may not be able to directly establish a connection with another agent in a different domain for some reasons. In such cases, the AGw needs to have routing management function. The routing management function generates

dynamic routing entries based on Agent information advertised by other AGws, and guides message forwarding between cross-domain Agents—essentially acting as a "cross-domain traffic commander" for the Agent ecosystem.

* Security Protection: Agent communication may involve sensitive data (such as user privacy, core business data), and malicious external Agents may attempt to forge identities, steal data, or launch attacks. The AGw serves as a core barrier for security protection. The AGw should supports the identity verification of Agents, data encryption, and access control for agents.

* Traffic Control: If multiple Agents simultaneously send discovery requests to the AGw, or concentrate on initiating collaboration requests after finding target Agents, the AGw may become overloaded or the target Agents may crash. The AGw should impose traffic limits on discovery requests. For example, it sets a cap on the number of discovery queries per Agent per minute to prevent malicious Agents from occupying resources with frequent queries.

* Dual Translation (Protocol and Language): Agents may face two key obstacles to direct communication: inconsistent technical protocols and different natural language expressions. The AGw’s translation function addresses both issues simultaneously to ensure seamless information transmission. On the one hand, it automatically identifies the communication protocol (e.g., HTTP, MQTT, gRPC, or custom protocols) and data format (e.g., JSON, Protobuf) used by the sending Agent, then converts them into the standard protocol/format supported by the receiving Agent. On the other hand, it handles natural language differences between Agents (e.g., Agent X uses English to send "Book a twin room" while Agent Y only understands Chinese). The AGw automatically translates the content into the target language, ensuring the receiving Agent accurately grasps the semantic intent without additional language adaptation.

3. Interactions of Agents Networking Architecture

3.1. Interactions between Agents and ACAs

3.1.1. Certificate Application

This is the most fundamental first interaction between an agent and an ACA. Its purpose is to enable the Agent to pass the ACA’s verification and obtain a digital certificate that proves its legitimacy. It includes three steps:

* Application request: The Agent sends a certificate application to the CA, along with key supporting materials. These materials include identity-verifying information (such as the qualification documents of its affiliated organization, the identity of its developer), the Agent's own attributes (such as function type, permission scope, skills), and a public key for subsequent encrypted communication (the Agent retains the corresponding private key and does not disclose it to external parties).

* Application verification: After receiving the application, the ACA checks the authenticity and compliance of the materials. For example, it verifies whether the organization's qualifications are valid, whether the public key meets encryption standards, and whether the Agent's functions fall within permitted scopes. If no issues are identified, the application is approved.

* Certificate generation: Upon approval, the ACA generates a digital certificate containing specific information. This information includes the Agent's identity details, public key, certificate validity period, and the ACA's digital signature (used to prove the certificate is legally issued by the CA). The CA then sends this certificate to the applying Agent.

3.1.2. Certificate Management

A certificate is not permanently valid. The Agent must cooperate with the ACA to complete operations such as renewal and revocation, ensuring the "digital identity" remains valid and secure.

* Certificate status query: The Agent can send a query request to the ACA at any time to confirm the current status of its certificate— such as whether it is still within the validity period or has been revoked due to abnormal circumstances. This allows the Agent to identify and resolve certificate-related issues promptly (e.g., applying to the CA for recovery if the certificate was incorrectly revoked).

* Certificate renewal: When the certificate is about to expire, the Agent proactively sends a renewal request to the ACA, along with the soon-to-expire old certificate. If its identity information (such as affiliated organization, permissions) has not changed, the application materials can be simplified. After the ACA approves the request, it generates a new certificate with an updated validity period and sends it to the Agent. The Agent replaces the old certificate with the new one to ensure the identity credential remains valid.

* Certificate revocation application: If the Agent experiences issues such as private key leakage, permanent deactivation of functions, or deregistration of its affiliated organization, the Agent (or its administrator) sends a certificate revocation request to the ACA to invalidate the currently held certificate. After confirming the request's legitimacy, the ACA adds the certificate to the "Certificate Revocation List (CRL)" to prevent the invalid certificate from being misused by others.

3.2. Interactions between Agents and ARSs

3.2.1. Registration

An Agent must first complete registration with the ARS before it can be discovered by other Agents and participate in collaboration. This is the foundational interaction between the two parties. It includes three steps:

* Registration request: The Agent sends a registration request to the ARS, accompanied by its information. This includes its identifier (e.g., Agent name), description of core capabilities (e.g., "hotel booking," "data format conversion"), supported communication protocols/languages (e.g., HTTP, Chinese), and credentials for identity verification (e.g., a digital certificate issued by an ACA).

* Registration verification: The ARS verifies the legitimacy of the Agent's identity (e.g., checking if the CA certificate is valid) and confirms the completeness of the registration information. Upon successful verification, the ARS creates a profile for the Agent in its "Agent Database," recording key data such as the Agent's capabilities, online status, and communication address to complete the registration.

* Registration response: The ARS feeds back results ("registration successful" or "registration failed," e.g., due to failed identity verification or missing information) to the Agent.

3.2.2. Information management

After registration, the Agent needs to continuously interact with the ARS to update its status and information, preventing the ARS's profile from becoming "outdated" and causing other Agents to match invalid resources.

* Information update: If an Agent upgrades its capabilities or changes its communication address (e.g., IP change due to server migration), it needs to proactively send an "information update

request" to the ARS with the updated content. After review, the ARS updates the corresponding profile information to ensure the accuracy of subsequent query results.

* Agent deregistration: When the Agent needs to go offline permanently (e.g., function deactivation, dissolution of the affiliated entity), it sends a "deregistration request" to the ARS with the resource ID. Upon receipt, the ARS marks the Agent's profile as "deregistered" or deletes it from the "discoverable resource pool" to avoid invalid queries.

3.2.3. Agents Discovery

This is the core service provided by ARS for agents, which helps them quickly identify target agents that can solve problems through multi-dimensional filtering, avoiding random searches, it includes two steps.

* Discovery request: If an Agent needs a collaborator with agents supporting specific capabilities, it sends a discovery request to the ARS, specifying the capabilities and associated other requirements.

* Discover response: Based on the request criteria, the ARS filters matching target Agents from the "Agent Database" and feeds back the agent information to Agent A, if the target Agents are in other domains, the returned communication address should be the AGw addresses of those domains.

3.3. Interactions between Agents and AGws

When an agent needs to communicate with an agent in another domain, the Communication sessions need to go through the AGw (When the two agents are within a single domain, whether a communication session needs to go through a AGw can be determined as requirements).

Interactions between an Agent and an AGw revolve around two core scenarios: status maintenance and collaboration support.

3.3.1. Status Maintenance

During the Agent's operation, ongoing interactions between the two parties are required to ensure stable and secure collaboration, primarily involving status synchronization and exception handling.

* Agent access: An agent should send an access request to the AGw to make it reachable by agents in other domains. The AGw reviews its information and verifies its signature. If the verification passed, then the AGw will create a record for it and advertise the agent information to other AGws.

* Status synchronization: After accessed, the Agent regularly sends "heartbeats" (e.g., every 30 seconds) to the AGw to synchronize its online status (e.g., "running normally," "temporarily offline"). If the AGw does not receive a "heartbeat" for an extended period, it marks the Agent’s status as "offline" and temporarily removes it from the collaborative list, and synchronizes this information to other potentially associated AGw to prevent other Agents from initiating invalid requests.

* Deregistration request: If an Agent needs to stop services, it proactively sends a deregistration request to the AGw. After receiving the request, the AGw deletes the Agent’s record, and synchronizes this information to other potentially associated AGws to complete the exit process.

3.3.2. Collaboration Support

When an Agent needs to collaborate with other Agents, the AGw provides support for "removing communication barriers". It is composed by two phases.

* Transformation: When the AGw receives the message send by the request agent with a translation request, the AGw first decapsulates the received packet, extracting the original content from the message. Then, it performs necessary adaptations, such as translating the content into the target language or converting the data structure to match the target protocol. Finally, the AGw re-encapsulates the adapted content into the target Agent’s protocol format and forwards it.

3.4. Interactions between AGws and AGws

Interactions between an AGw and an AGw revolve around four core phases: AGw interconnection, resources synchronization, status synchronization, and relay forwarding.

3.4.1. AGw Interconnection

This phase completes mutual identity verification and initial status synchronization between AGws.

* Interconnection request: The initiating AGw sends an interconnection request(the request and) to the receiving AGw, enclosing core information: its unique AGw ID, supported protocols, identity certificate, and current basic operational status.

* Interconnection response: The receiving AGw verifies the initiating AGw's identity and protocol compatibility. If valid, it feeds back its own basic information and records AGw A's initial information in its "AGw Neighbor List". If the verification failed, then it rejects the request.

3.4.2. Resources Synchronization

Resource synchronization and status maintenance focuses on sharing static/dynamic information of registered Agents, ensuring AGws can accurately locate potential collaboration partners across domains.

* Resource synchronization: AGws exchange the summary of registered agents' information, including Agent ID, core capabilities (e.g., "flight booking", "real-time translation"), and supported communication protocols/languages (e.g., gRPC, Chinese). The Synchronization take place whenever a new AGw peer is established or the registered agents' key information changes.

3.4.3. Status Maintenance

The Status Maintenance of AGws is used to ensure the peer AGws are online and reachable.

* Status synchronization: When the interconnection is established, the AGws regularly sends "heartbeats" (e.g., every 30 seconds) to the peer AGws to synchronize its online status (e.g., "running normally," "temporarily offline"). If the AGw does not receive a "heartbeat" for an extended period, it marks the AGw's status as "offline" and temporarily removes it from the peer list, and set the "applicability" attribut of agents registered at that peer AGw as inapplicable.

* Disconnection request: If an AGw needs to stop services, it proactively sends a disconnection request to its peer AGw. After receiving the request, the peer AGw deletes the AGw from its peer list, and remove the Agents registered at that peer AGw from the "Agent Database".

3.4.4. Relay Forwarding

Relay Forwarding leverages pre-synchronized resource and status data to enable seamless data transmission between cross-AGw Agents.

* When an AGw receives a packet from its agents to an agent in another domain, it will decapsulates the packet and get the "agent ID" of the target agent. Then it looks up the routing table to get the target AGw and forward the packet to it (during the forwarding, the packet will be encapsulated to adapt to the target agent).

3.5. Interactions between ARSs

Interactions between ARS focus on "breaking resource silos". Through cross-ARS information synchronization, they enable Agent resources managed by different ARSs to be globally discoverable and collaborable in a trusted manner. The detailed interactions between ARS can be quit different among different ARS forms, it depends on the implement manner of ARS.

Editors’ note: typical interactions between ARS will be provide in future.

4. Operational and Manageability Considerations

TBD

5. Security Considerations

TBD

6. IANA Considerations

This document has no IANA actions.

7. Acknowledgements

TBD

8. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

Authors' Addresses

Li Zhang
Huawei Technologies
No. 156 Beiqing Road
Beijing
China
Email: zhangli344@huawei.com

Bing Liu
Huawei Technologies
No. 156 Beiqing Road
Beijing
China
Email: leo.liubing@huawei.com

Nan Geng
Huawei
Beijing
China
Email: gengnan@huawei.com

ZiHang Wang
Independent
Beijing
China
Email: 1639283202@qq.com