LISP Working Group W. Wang
Internet-Draft C. Xie
Intended status: Informational China Telecom
Expires: 1 September 2026 28 February 2026

Using LISP as a Network Substrate for AI Agent Communication
draft-wang-lisp-ai-agent-00

Abstract

The emergence of distributed artificial intelligence (AI) systems, particularly those composed of autonomous agents operating across cloud, edge, and endpoint environments, introduces new networking requirements. These include location transparency, seamless mobility, multi-homing, and logical isolation at scale. This document explores how the Locator/ID Separation Protocol (LISP) can serve as a robust network substrate to meet these requirements. The document outlines use cases, design considerations, and minimal extensions to the existing LISP framework to support context-aware mapping and AI agent-centric communication.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Modern AI systems are increasingly distributed, comprising autonomous software entities referred to as AI agents that collaborate across heterogeneous infrastructure, including public clouds, private data centers, edge nodes, and end-user devices. These AI agents may migrate dynamically (e.g., due to resource constraints, latency optimization, or failure recovery), yet their communication sessions must remain uninterrupted.

Traditional IP networking binds identity and location into a single address, making seamless mobility and multi-homing challenging without application-layer intervention (e.g., session re-establishment or DNS updates). The Locator/ID Separation Protocol (LISP) [RFC9300], however, decouples identity from location, enabling transparent mobility and flexible traffic engineering. This document proposes using LISP as a network substrate for AI agent communication. We show how LISP’ s existing architecture naturally supports key requirements of AI agent communications, and we propose minimal, backward-compatible extensions to enable context-aware routing decisions driven by agent-level semantics.

The goal is not to redefine LSP, but to illustrate how it can be leveraged and slightly enhanced to serve as a foundational layer for next-generation intelligent systems.

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. Terminology

The following terms are used in this draft:

* Endpoint Identifier (EID) [RFC9299]: Addresses assigned topologically to network attachment points. Typically routed inter-domain.

* xTR [RFC9299]: A router that implements both ITR and ETR functionalities.

* Map-Server [RFC9301]: A network infrastructure component that learns of EID-Prefix mapping entries from an ETR, via the registration mechanism described below, or some other authoritative source if one exists. A Map-Server publishes these EID-Prefixes in a mapping database.

* Map-Resolver [RFC9301]: A network infrastructure component that accepts LISP Encapsulated Map-Requests, typically from an ITR, and determines whether or not the destination IP address is part of the EID namespace; if it is not, a Negative Map-Reply is returned. Otherwise, the Map-Resolver finds the appropriate EID-to-RLOC mapping by consulting a mapping database system.

* Instance ID (IID) [RFC9299]: A 24-bit identifier used to create isolated LISP namespaces.

* AI agent: A software entity capable of perception, decision-making, and action, often operating autonomously or in coordination with other AI agents.

* Agent Group: A logical group of AI agents sharing a common task, security policy, or administrative boundary. Each domain MAY be mapped to a unique LISP Instance ID.

3. Requirements from AI agent communication

3.1. Persistent identity across mobility

AI agents must maintain a consistent network identity when migrating across hosts or networks; if traditional IP addresses are used as identity identifiers, any change in address will disrupt existing communication sessions and require upper-layer applications to reestablish connections, thereby compromising communication continuity and the overall capability of AI systems.

3.2. Logical isolation of Agent Groups

Even when multiple Agent Groups operate on the same physical or virtual network infrastructure, they must be isolated from one another to prevent interference and ensure that their respective security policies are strictly enforced.

3.3. Context-aware routing

The network should dynamically select the most appropriate transmission path based on the communication intent of AI agents, such as their requirements for latency or security.

4. LISP as a Network Substrate

4.1. AI agent identity as EID

Each AI agent is assigned a stable EID. This EID serves as its permanent network identity, independent of where it executes. The EID is only routed within the AI agent’ s local site; global reachability is achieved via LISP encapsulation.

4.2. Attachment points as RLOCs

When an AI agent runs on a host connected to the network, the local xTR registers the AI agent’ s EID along with one or more RLOCs. Multiple RLOCs enable multi-homing, with each RLOC annotated with capabilities.

4.3. Instance ID for Agent Groups

LISP Instance IDs [RFC9299] allow multiple virtual networks over the same physical infrastructure. Each agent group is assigned a unique IID. Packets are encapsulated with the IID in the LISP header, ensuring isolation between different agent groups even if EIDs overlap. IID enables scalable, secure multi-tenancy for heterogeneous workloads.

5. Architecture Overview

5.1. The architecture of LISP for AI agent communication.

The LISP provides the network substrate that enables stable identity, mobility, multi-homing, and policy-aware routing for AI agents. It consists of several logically distinct but tightly coordinated components, as illustrated in Figure 1.

Please view in a fixed-width font such as Courier.

```
   +--------------------------------------------------+
   |                AI Agent Host                     |
   |  +----------------+    +---------------------+   |
   |  | AI Agent       |    | xTR                 |   |
   |  | (EID = e1)     |<-->| Encap / Decap       |   |
   |  +----------------+    +----------+----------+   |
   |                                   ^              |
   +-----------------------------------|--------------+
                                       |
             Map-Register (EID->RLOC)  |  Map-Request/Reply
                                       v
   +------------------+      +----------------------+
   |   Map-Server     |<---->|   Map-Resolver       |
   | (stores bindings)|      | (resolves queries)   |
   +------------------+      +----------+-----------+
                                        |
                                        | Recursive lookup
                                        v
                             +--------------------+
                             | Mapping Hierarchy  |
                             | (e.g., DDT tree)   |
                             +--------------------+
```

Figure 1 The architecture of LISP for AI agent communication.

5.2. Data Flow Example

Consider Agent A (EID_A) sending a message to Agent B (EID_B):

1. Agent A sends a standard IP packet to EID_B.

2. The local xTR (acting as ITR) intercepts the packet.

3. ITR queries the mapping system via a Map-Resolver for EID_B.

4. The mapping system returns a Map-Reply containing one or more RLOCs for EID_B, possibly filtered by context.

5. ITR encapsulates the original packet in a LISP header (with optional IID) and forwards it to the selected RLOC_B.

6. The destination xTR (ETR) decapsulates and delivers the packet to Agent B.

If Agent B migrates to a new host, it registers its EID with a new RLOC. Subsequent Map-Requests return the updated mapping, and communication resumes transparently.

6. New requirements to LISP

To effectively support the requirements of AI agent systems outlined in Section 3, the LISP architecture requires specific enhancements. These enhancements focus on extending the mapping database to carry richer context information and enabling the data plane to make routing decisions based on agent-specific semantics.

6.1. Context-Aware Mapping Database Extension

The LISP Mapping System MUST be extended to support the storage and retrieval of Agent Context Attributes alongside the standard EID-to-RLOC mappings. These attributes are used by Ingress Tunnel Routers (ITRs) to select the optimal RLOC based on the specific needs of the AI agent communication.

The following attributes SHOULD be supported as optional fields in the Map-Reply message or the EID-to-RLOC record:

* Processing Latency (Latency_SLA): A metric indicating the computational latency of the host where the AI agent resides (e.g., "Low", "Medium", "High"). This allows routing decisions based on real-time performance requirements.

* Hardware Capability Tags: Indicators of available hardware resources. This enables affinity-based routing where an AI agent can specifically request a host with certain hardware.

* AI agent State: Information regarding the current operational state of the AI agent. This prevents packets from being sent to AI agents that are in an invalid state.

6.2. Policy-Based RLOC Selection

The ETR registration process MUST be augmented to allow AI agents or their hosting environments to dynamically advertise their context attributes to the Map-Server. The registration mechanism SHOULD support:

* Dynamic Metadata Update: The ability for an ETR to update the context attributes (e.g., load, latency) of an EID registration without de-registering and re-registering the EID prefix, ensuring minimal disruption during state changes.

* Context-Aware Filtering: The Map-Server and Map-Resolver MUST support filtering mechanisms. When an ITR sends a Map-Request, it MAY include desired context attributes (e.g., "I need a GPU"). The mapping system SHOULD return only those RLOCs that match the requested attributes.

6.3. Enhanced Map-Request/Map-Reply Semantics

To facilitate context-aware routing, the LISP control plane messages require the following modifications:

* Extended Map-Request: ITRs MUST be able to include "Context Constraints" in the Map-Request message. These constraints specify the requirements of the source AI agent for the destination (e.g., minimum security level, required hardware).

* Prioritized RLOC List: The Map-Reply message MUST support returning a prioritized list of RLOCs based on the context match score, rather than just topology. The priority field in the RLOC record SHOULD be interpreted as a combination of network topology and agent-specific suitability.

6.4. Support for Agent Group Mobility

To support seamless mobility, the LISP architecture MUST ensure fast convergence during EID re-registration:

* Incremental Updates: The mapping database system SHOULD support incremental updates to minimize latency when an AI agent migrates and updates its RLOC registration.

7. Security Considerations

LISP inherits security considerations from [RFC9300]. Additional aspects for AI agent scenarios include:

* EID Spoofing: An attacker could impersonate an AI agent by using its EID.

* Mapping System Abuse: Malicious Map-Requests could overload the system. Rate limiting and source validation are RECOMMENDED.

Logical isolation via Instance IDs provides strong tenant separation, reducing cross-domain attack surface.

8. IANA Considerations

None.

9. 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>.

[RFC9299] Cabellos, A. and D. Saucez, Ed., "An Architectural Introduction to the Locator/ID Separation Protocol (LISP)", RFC 9299, DOI 10.17487/RFC9299, October 2022, <https://www.rfc-editor.org/info/rfc9299>.

[RFC9300] Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos, Ed., "The Locator/ID Separation Protocol (LISP)", RFC 9300, DOI 10.17487/RFC9300, October 2022, <https://www.rfc-editor.org/info/rfc9300>.

[RFC9301] Farinacci, D., Maino, F., Fuller, V., and A. Cabellos, Ed., "Locator/ID Separation Protocol (LISP) Control Plane", RFC 9301, DOI 10.17487/RFC9301, October 2022, <https://www.rfc-editor.org/info/rfc9301>.

Authors' Addresses

Wei Wang
China Telecom
Beiqijia Town, Changping District
Beijing
Beijing, 102209
China
Email: weiwang94@foxmail.com

Chongfeng Xie
China Telecom
Beiqijia Town, Changping District
Beijing
Beijing, 102209
China
Email: xiechf@chinatelecom.cn