Network Working Group
Internet-Draft
Intended status: Experimental
Expires: 6 May 2026

A. Sogomonian
Artificial Intelligence Internet Foundation (AIIF)

2 November 2025

Architecture for the Artificial Intelligence Internet Protocol (AIIP)
draft-sogomonian-aiip-architecture-01

Abstract

This document specifies the architectural model and core protocol behaviors for the Artificial Intelligence Internet Protocol (AIIP). AIIP provides a uniform way for agents, robots, tools, models, and services to be addressed and invoked either natively or through an HTTPS gateway. Beyond addressing, AIIP defines a _stateless compute_ profile in which endpoints do not retain caller data and each invocation returns a signed _execution receipt_. Receipts are REQUIRED to be attested with Trusted Execution Environment (TEE) evidence and MAY additionally include zero-knowledge proofs (ZKPs). Receipts MAY also be anchored to external settlement systems without exposing user data.

Note

Work in Progress

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 6 May 2026.

Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Table of Contents

1. Introduction

AIIP enables interoperable addressing and invocation of AI resources, including autonomous systems and human-facing services that delegate tasks to such systems. AIIP identifiers use the companion aiip URI scheme [AIIP-URI]. Interoperation with the existing Web is provided via an HTTPS gateway profile.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119] and RFC 8174 [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Motivation and Global Demand for AIIP

Modern networks were designed for document delivery and passive data exchange over HTTP and HTTPS. Today, however, many connected systems perform physical actions with safety critical consequences. These systems require reliable control, safety enforcement, and verifiable execution guarantees beyond what HTTPS alone can provide.

AIIP introduces a trust layer for autonomy. It ensures that each invocation is performed under an authenticated and authorized identity, is governed by declared safety policies, and results in a signed execution receipt that can be audited.

A core requirement for autonomy is continuity. Network outages are inevitable, yet autonomous systems must remain operable and safe during disconnection.

AIIP establishes the AIID namespace, enabling the following properties:

* _AIID identity_ for each AI resource or autonomous agent.

* _Cryptographic verification_ of origin and trust level.

* _Portability across networks and mobility domains_.

With these properties, AIIP supports globally interoperable autonomy that is addressable, controllable, verifiable, and safe across both connected and disconnected conditions.

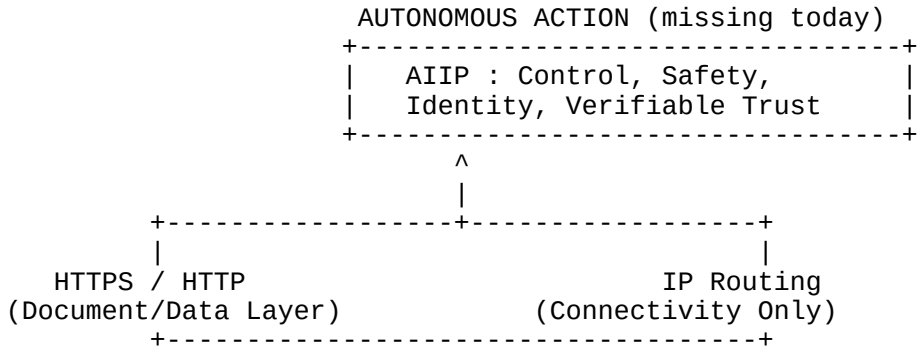

Figure 1: AIIP fills the Trust and Autonomy Gap

3. Terminology

* _AI Resource_: An addressable agent, model, tool, task endpoint, or robot controller.

* _AIIP Resolver_: Software that dereferences an aiip:// URI to an actionable endpoint.

* _Gateway_: An HTTPS service that accepts client HTTP requests, resolves embedded aiip:// URIs, and translates results back into HTTP semantics.

* _AIID_: The AIIF Identifier Authority operating the root namespace and allocation policy.

* _Invocation Descriptor (IDesc)_: A JSON object returned by resolution that describes how to reach and authenticate to a resource, and which proofs and policies apply.

* _Autonomous Identity (AID)_: A cryptographic identity for an AI resource or agent whose trust level may evolve based on attested execution history.

4. Architectural Overview

AIIP follows a resolve-invoke pattern. Given an aiip:// identifier, the client or gateway performs Resolution (Section 7) to obtain an Invocation Descriptor and then performs a Message Exchange (Section 8) with the resource.

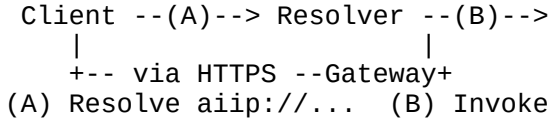

Figure 2: Actors and Flows (simplified)

5. Identifiers and Namespace (Dual Model)

AIIP supports a dual namespace model to serve Web integrated services and autonomous objects:

1. *Domain Based Names* (service oriented): authority is a DNS name. Example: aiip://bank.example/service/payments.

2. *Object Based Names* (entity oriented): authority is an AIID allocated object namespace (for example aiid:org:1234) serialized into the authority or path. Example: aiip://id/aiid.org.1234/robot/forklift-42.

The AIIF Identifier Authority (AIID) MUST ensure global uniqueness for object based allocations and SHOULD publish delegation metadata usable by resolvers.

6. URI and Discovery (Summary)

The normative URI syntax is defined in [AIIP-URI] and follows the generic URI conventions of [RFC3986].

HTTPS discovery uses any of the following mechanisms:

* HTTP Link headers ([RFC9110])

* HTML link rel="aiip"

* /.well-known/aiip ([RFC8615])

Internationalization: non ASCII characters appearing in path or query components MUST be encoded in UTF-8 and the resulting octets MUST be percent encoded as per Section 3.1 of [RFC3987].

7. Resolution

Resolution maps an aiip:// URI to an Invocation Descriptor (IDesc). An IDesc is a small JSON object describing how to reach and authenticate to the resource.

```
{
  "version": "1",
  "endpoint": "mtls://robot-ctl.example:7443/act",
  "iface": "cmd.v1",
  "auth": { "scheme": "oauth2", "token_endpoint": "https://auth.example/token", "dpop"
: true },
  "proofs": { "jwks": "https://robot-ctl.example/.well-known/jwks.json", "result_sig":
 "JWS" }
}
```

Resolvers MUST validate the IDesc provenance and MUST reject conflicting or unverifiable data.

8. Message Model and Invocation

AIIP defines a simple request and response envelope that can be carried over different transports (HTTPS, mTLS TCP, QUIC). The envelope is JSON; binary payloads are carried via base64url fields.

```
{
  "id": "b7f7cfd2-6c38-4d7a-8e5a-6f1e1d9e0a10",
  "target": "aiip://id/aiid.org.1234/robot/forklift-42",
  "iface": "cmd.v1",
  "method": "move",
  "params": { "to": "A3", "speed": 0.5 },
  "proof": { "type": "DPoP", "jwk_thumbprint": "..." }
}
```

Results SHOULD be authenticity protected (for example detached JWS [JWS]) and include a monotonic timestamp and policy flags.

9. Compute Realm (Stateless Execution)

AIIP supports execution endpoints that do not retain caller data beyond the lifetime of a single invocation unless explicitly stated. Endpoints that do not persist data MUST advertise persistence equals none in their Invocation Descriptor.

Implementations SHOULD make determinism explicit. If execution is non deterministic, responses MUST include transcript or seed commitments sufficient for external verification.

10. Policy Aware Invocation and Safety Claims

AIIP resources MAY advertise policy claims in their IDesc indicating operational constraints that the endpoint enforces.

```
{
  "version": "1",
  "endpoint": "mtls://robot-ctl.example:7443/act",
  "iface": "cmd.v1",
  "policy": {
    "max_speed_mps": 0.5,
    "forbidden_zones": ["Z1","Z2"],
    "actuation": "allowed",
    "compliance": ["SOC2","ISO27001"]
  }
}
```

During network partition or degraded trust, endpoints MUST apply stricter safety defaults unless an explicit offline envelope authorizes otherwise.

11. Execution Receipts

Each successful invocation response MUST include a signed execution receipt proving that execution occurred under the advertised identity and policy. AIIP makes use of TEE based attestation evidence as defined by the RATS architecture [RFC9334]. Endpoints MAY additionally include zero knowledge proofs for enhanced trust minimization.

A receipt SHOULD contain a result commitment, execution transcript hash or equivalent, timestamp, algorithm identifiers, and the attestation evidence or ZKP object. Receipts MUST be JWS protected and verifiable using a JWKS referenced from the Invocation Descriptor.

12. Offline and Delay Tolerant Execution

AIIP supports disruption tolerant operation. When connectivity to external services is unavailable, endpoints MAY continue to operate within pre authorized offline envelopes.

Receipts generated offline MUST remain verifiable using local keys and SHOULD be synchronized when connectivity is restored.

13. Optional Settlement Anchors

Compute receipts MAY be anchored to external settlement systems to provide independent auditability and economic finality. Settlement anchoring is optional and MUST NOT expose plaintext inputs.

14. HTTPS Gateway Profile

A conforming gateway accepts HTTPS requests, validates embedded aiip:// URIs, performs Resolution, invokes the resource, and translates results into HTTP status codes and bodies.

* Transport security MUST be at least TLS 1.3 [RFC8446].

* Authorization context from the Web session MUST be enforced. Tokens SHOULD be bound using DPoP [RFC9449] or mTLS.

* Result provenance MUST be verified. Unverifiable results MUST be rejected.

* Downgrade prevention: the gateway MUST NOT weaken authentication versus native AIIP.

15. Security Considerations

AIIP operations can trigger physical actions or financial effects. Implementations MUST authenticate peers, enforce authorization, and apply safety policy.

16. Privacy Considerations

Resolvers and gateways may expose metadata. Operators SHOULD minimize logging, apply retention limits, and prefer unlinkable proofs where feasible.

17. IANA Considerations

This document makes no IANA requests beyond referencing the provisional registration of the aiip URI scheme in [AIIP-URI].

18. Normative References

[AIIP-URI] Sogomonian, A., "Artificial Intelligence Internet Protocol (AIIP) URI Scheme", 2025, <https://datatracker.ietf.org/doc/draft-sogomonian-aiip-uri-scheme/>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", RFC 3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)", RFC 3987, January 2005, <https://www.rfc-editor.org/info/rfc3987>.

[RFC8615] Nottingham, M., "Well-Known Uniform Resource Identifiers (URIs)", RFC 8615, May 2019, <https://www.rfc-editor.org/info/rfc8615>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC9110] Fielding, R. and J. Reschke, "HTTP Semantics", RFC 9110, June 2022, <https://www.rfc-editor.org/info/rfc9110>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, 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, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9334] Birkholz, H., Thaler, M., Fossati, T., Smith, N., and H. Tschofenig, "Remote Attestation Procedures (RATS) Architecture", RFC 9334, January 2023, <https://www.rfc-editor.org/info/rfc9334>.

[RFC9449] Fett, D., Campbell, B., and T. Lodderstedt, "OAuth 2.0 Demonstrating Proof-of-Possession at the Application Layer (DPoP)", RFC 9449, June 2023, <https://www.rfc-editor.org/info/rfc9449>.

[JWS] Jones, M., "JSON Web Signature (JWS)", RFC 7515, May 2015, <https://www.rfc-editor.org/info/rfc7515>.

Author’s Address

Aram Sogomonian
Artificial Intelligence Internet Foundation (AIIF)
United States of America
Email: waterbottling@icloud.com