Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 19 October 2026

Z. T.
INUID Research Group

17 April 2026

Internet Unique ID (INUID) Protocol Specification
draft-inuidgroup-inuid-00

Abstract

This document specifies the Internet Unique ID (INUID) protocol, a 128-bit network-layer protocol designed to decouple endpoint identity from topological location. INUID introduces a hierarchical addressing model, fixed-size header layout, mandatory cryptographic source verification, and explicit support for mobility and multihoming without requiring upper-layer session re-establishment.

INUID is intended to address several persistent weaknesses in contemporary internetworking, including inadequate source authenticity, unsustainable routing table growth, and the operational complexity of legacy transition mechanisms. The protocol is designed for efficient hardware forwarding, strong anti-spoofing guarantees, and scalable inter-domain routing behavior.

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). This document is not an Internet Standard and is published for examination, implementation testing, and discussion.

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 19 October 2026.

Table of Contents

1. Introduction

The modern Internet continues to rely on an addressing model in which endpoint identity and routing location are fused into a single value. While this model enabled the rapid growth of packet networking, it creates structural limitations in environments that demand strong endpoint authentication, efficient mobility, persistent sessions, scalable multihoming, and bounded global routing state.

The Internet Unique ID (INUID) protocol addresses these issues by separating identity from location at the network layer. In INUID, a packet carries both a stable Identity Prefix and a routable Locator. The identity identifies the communicating endpoint, while the locator identifies the endpoint’s current topological attachment point.

This separation allows transport sessions, access policies, and security associations to bind to stable identities, while the forwarding plane operates only on aggregated locators. As a result, network mobility and path changes can occur without breaking the higher-layer communication state.

INUID additionally requires cryptographic proof of source identity at the network edge. This design makes unauthenticated source spoofing substantially more difficult and allows access networks to enforce identity ownership before forwarding traffic into the core.

2. 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 [RFC2119] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology

* *Identity Prefix:* A 64-bit stable identifier assigned to an endpoint, endpoint group, or delegated identity namespace.

* *Locator:* A 48-bit routing value indicating the topological region through which traffic for an endpoint is forwarded.

* *Tag:* A 16-bit field used for attachment-specific demultiplexing, temporary sub-identities, or local policy handling.

* *CSV:* Cryptographic Source Verification, the validation mechanism by which a router confirms that a sender is authorized to use a claimed source identity.

* *IA:* Identity Advertisement, the local-link mechanism used by endpoints to announce identity material and capabilities to an access router.

* *Edge Router:* A router at an administrative boundary or access domain that validates endpoint identity and authorizes locator use.

* *Transit Router:* A router in the forwarding core that routes packets based primarily on Locator values.

4. Design Goals

INUID has the following design goals:

1. Provide a stable network-layer identifier that is independent of network attachment point.

2. Constrain global routing growth by advertising only locator reachability in the routing core.

3. Prevent unauthenticated source address spoofing through mandatory source verification at the edge.

4. Support session continuity during locator change events such as mobility and multihoming failover.

5. Maintain a simple base header suitable for line-rate forwarding in software and hardware implementations.

6. Permit extensibility through explicit extension headers rather than overloading the fixed header format.

5. Architecture Overview

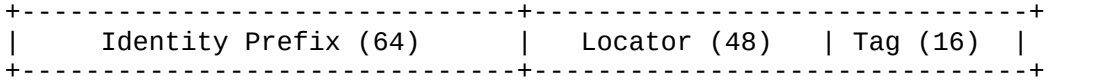
INUID uses a structured 128-bit endpoint representation composed of two logical halves. The first half is the Identity Prefix, which remains stable over time and is cryptographically bound to an authorized entity. The second half is a routable suffix composed of a Locator and Tag.

Packets are forwarded through the core using only the Destination Locator. The Destination Identity Prefix is retained end-to-end but is not required for transit routing decisions. This division allows the forwarding plane to preserve aggregation while the endpoint identity remains meaningful to hosts, access routers, policy engines, and security systems.

INUID defines a mandatory edge-validation model. A packet entering an INUID administrative domain from an attached endpoint MUST be subjected to Cryptographic Source Verification before the packet is forwarded beyond the trust boundary. Transit routers are not required to re-verify identities for each hop, though domains MAY apply additional policy checks where appropriate.

6. Addressing Model

An INUID address consists of the following conceptual fields:



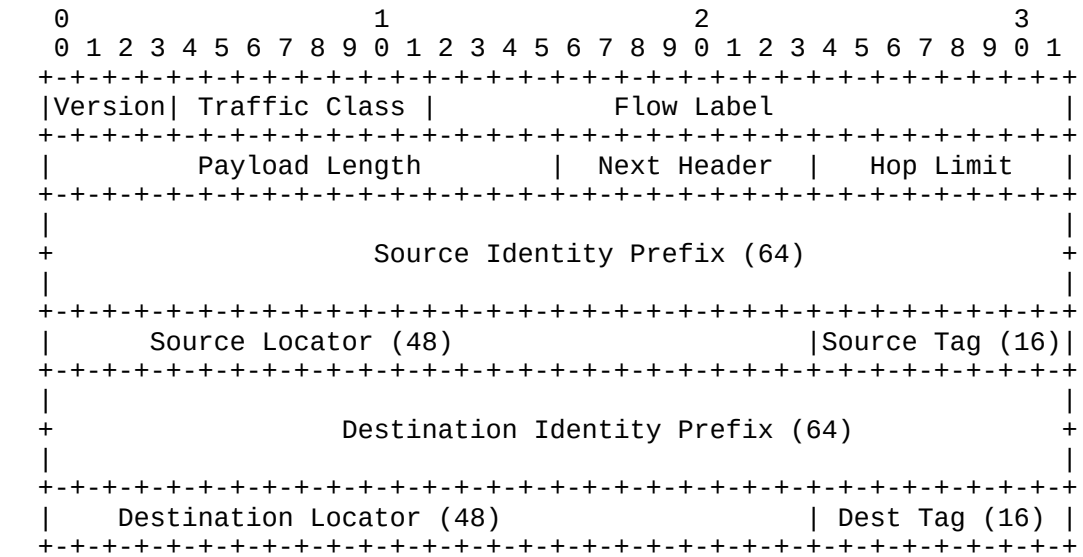
The Identity Prefix identifies the logical endpoint or namespace. It is intended to remain stable across movements, renumbering events, and access-network changes, unless replaced due to explicit administrative or cryptographic policy.

The Locator identifies a routing region and is used by forwarding systems to deliver the packet to the correct destination domain or attachment point.

The Tag provides local disambiguation. It MAY identify a temporary attachment instance, service endpoint, virtual interface, or short-lived session-scoping construct. The Tag MUST NOT be interpreted as a substitute for cryptographic proof of identity ownership.

7. INUID Base Header Format

The INUID base header is a fixed 40-byte structure optimized for alignment with common forwarding and parsing architectures.



7.1. Version

The Version field identifies the packet as using the INUID protocol. This specification requests allocation of version value 9. Nodes receiving an unsupported version value MUST discard the packet.

7.2. Traffic Class

The Traffic Class field conveys differentiated forwarding treatment, congestion behavior, or domain-specific quality-of-service intent. Routers SHOULD process this field consistently with existing operational practice for service differentiation.

7.3. Flow Label

The Flow Label identifies packets that belong to a common forwarding or treatment flow. Senders SHOULD use a stable Flow Label for packets that require path consistency or efficient per-flow load distribution.

7.4. Payload Length

The Payload Length field indicates the number of octets following the INUID base header, including all extension headers and upper-layer payload data.

7.5. Next Header

The Next Header field identifies the immediately following extension header or upper-layer protocol. Unknown values MUST result in packet discard unless a local policy explicitly defines alternative handling.

7.6. Hop Limit

The Hop Limit is decremented by each forwarding node. A node that reduces the Hop Limit to zero MUST discard the packet and SHOULD generate an error indication subject to local policy.

7.7. Source and Destination Fields

The Source and Destination Identity Prefix fields identify the communicating entities. The corresponding Locator and Tag fields represent current topological reachability and local delivery context.

8. Extension Headers

INUID supports a chained extension-header model. Each extension header MUST begin with a Next Header field and a Header Length field unless otherwise specified by a future standards-track extension definition.

The following extension header categories are defined for INUID deployments:

8.1. Authentication Extension Header

The Authentication Extension Header carries the proof material needed for Cryptographic Source Verification. This header MAY include a key identifier, certificate reference, signature value, freshness token, replay-protection material, or a compact proof chain.

Packets originating from endpoint systems onto an INUID-native attachment link MUST include this header unless exempted by a tightly scoped local infrastructure policy.

8.2. Mobility Update Extension Header

The Mobility Update Extension Header carries authenticated locator update information. It is used when an endpoint changes attachment point while retaining the same Identity Prefix.

8.3. Telemetry Extension Header

The Telemetry Extension Header MAY carry optional domain-defined path or measurement metadata. Domains that do not support such processing MUST be able to ignore or remove this header according to policy.

8.4. Legacy Translation Context Header

The Legacy Translation Context Header MAY be used by transition gateways when additional context is required to interoperate with non-INUID networks.

9. Protocol Operations

9.1. Packet Origination

An originating node constructs a packet by selecting the Source Identity Prefix assigned to it, a currently valid Source Locator, and an optional Source Tag. The node then populates the corresponding destination fields using information obtained from local policy or a locator-resolution mechanism.

Before transmission, the sender or first-hop router MUST attach authentication material sufficient for CSV enforcement.

9.2. Hierarchical Routing

Transit routers MUST forward packets using the Destination Locator. Transit routing systems MUST aggregate reachability announcements according to locator allocation boundaries.

Identity Prefix values MUST NOT be globally propagated into the routing core as ordinary forwarding prefixes except in explicitly limited environments. This ensures that global forwarding state scales with locator allocation rather than the number of endpoints.

9.3. Cryptographic Source Verification (CSV)

INUID edge routers MUST perform CSV on traffic received from directly attached endpoints or subordinate domains before forwarding such traffic into the broader network.

CSV MUST confirm at least the following properties:

1. The sender possesses authority over the claimed Source Identity Prefix.

2. The proof material is bound to the packet, flow, or short-lived authorization context.

3. The claimed source is valid for the observed point of attachment or delegated routing relationship.

4. The proof is sufficiently fresh to prevent trivial replay attacks.

If verification fails, the packet MUST be silently discarded. Implementations SHOULD record counters and MAY generate local audit events, but they MUST avoid behavior that creates reflection or amplification opportunities.

9.4. Neighbor Discovery and Identity Advertisement

INUID replaces traditional ARP and NDP behavior with Identity Advertisement (IA). When an endpoint joins a local segment, it announces its Identity Prefix and supporting proof material to the first-hop router.

The first-hop router validates the advertisement and authorizes a Source Locator and Source Tag for use on that segment. IA messages MUST be link-local in scope unless relayed through an explicitly defined control-plane mechanism.

9.5. Mobility

An endpoint MAY change its Locator while retaining its Identity Prefix. When this occurs, the endpoint or its new first-hop router MUST generate a valid mobility update so that communicating peers or locator-resolution agents can refresh their reachability state.

Transport associations bound to the Identity Prefix SHOULD survive locator updates provided that freshness, authorization, and replay checks succeed.

9.6. Multihoming

A single Identity Prefix MAY be associated with multiple Locators. This allows an endpoint or site to maintain simultaneous reachability through multiple providers or attachment domains.

Locator selection MAY be based on preference, policy, performance, cost, or failover considerations. Core routers remain unaware of identity-level multihoming semantics and continue to forward only on Locator values.

10. Packet Processing Rules

10.1. Host Processing

An INUID host receiving a packet MUST verify that the Destination Identity Prefix matches a locally assigned identity or a delegated service identity. The host SHOULD also validate that the Destination Tag is acceptable within the local service context when tag-aware delivery is in use.

10.2. Router Processing

A forwarding node processing an INUID packet MUST perform the following checks in order:

1. Verify the base header length and packet completeness.
2. Verify that the Version field is supported.
3. Decrement the Hop Limit for transit forwarding.
4. Parse the Next Header chain according to local capability and policy.
5. Perform edge verification if the interface is subject to CSV enforcement.
6. Forward the packet based on the Destination Locator or deliver it locally if appropriate.

10.3. MTU and Fragmentation

This specification does not define in-network fragmentation by routers. Transit routers MUST NOT fragment INUID packets. Sources are expected to perform suitable packet sizing based on path characteristics and implementation guidance.

If a packet exceeds the outgoing path capability, a router SHOULD generate an implementation-specific Packet Too Large indication, subject to security and policy controls.

11. Identity-to-Locator Resolution

INUID requires a mechanism by which senders obtain a currently valid Destination Locator for a given Destination Identity Prefix. This document does not mandate a single universal resolution system, but any such mechanism MUST provide authenticity, freshness, and protection against unauthorized locator publication.

Resolution systems MAY be DNS-based, rendezvous-based, directory-based, or domain-delegated. Regardless of mechanism, a sender MUST be able to validate that the returned Locator is authorized for the requested Identity Prefix.

12. Transition and Compatibility

INUID is expected to coexist with IPv4 and IPv6 during a long transition period. To support compatibility, this specification defines a Legacy Translation range at ::FFFF:0:0/96.

Packets entering an INUID-native network from an IPv4-only domain MAY be mapped into this range by a translation gateway. Such mapped values MUST be treated as compatibility encodings rather than as proof of native INUID identity ownership.

Gateways MUST preserve a clear distinction between native cryptographically bound identities and translated legacy addresses. Implementations MUST NOT infer cryptographic identity from a legacy-mapped address alone.

13. Deployment Considerations

13.1. Incremental Deployment

INUID is designed to support staged deployment. An operator MAY first deploy INUID inside a single administrative domain, then extend CSV to access edges, and finally enable inter-domain locator exchange and mobility support.

13.2. Hardware Considerations

The fixed base header and bounded parser behavior make INUID suitable for implementation in forwarding ASICs, NPUs, and software fast paths. Implementations SHOULD optimize access to common identity-verification metadata and SHOULD support cryptographic offload where available.

13.3. Policy Considerations

Operators SHOULD define policy for identity delegation, acceptable cryptographic algorithms, locator aggregation boundaries, mobility authorization, extension-header limits, and telemetry visibility.

14. Manageability Considerations

Implementations SHOULD expose operational counters and logs for at least the following events:

* CSV successes and failures
* Identity Advertisement acceptance and rejection events
* Locator update acceptance and rejection events
* Unsupported Next Header or extension-header processing failures
* Hop Limit expiration events
* Legacy translation mappings and failures

Operational tools SHOULD support visibility by Identity Prefix, Locator, interface, and policy domain, subject to privacy and retention requirements.

15. Error Handling

Implementations SHOULD define an error signaling mechanism analogous to existing control-message frameworks. Such a mechanism SHOULD support notification for Hop Limit exhaustion, unsupported header types, validation failure diagnostics for trusted domains, and packet size violations.

Error signaling MUST be rate-limited and MUST NOT create reflection, amplification, or identity disclosure vulnerabilities.

16. Security Considerations

INUID is designed to improve network-layer security by requiring source identity validation at the edge. This significantly reduces the feasibility of forged-source attacks and provides a stronger foundation for attribution, filtering, and trust-domain enforcement.

However, the protocol introduces new security considerations that implementations and operators MUST address:

* *Verification Cost:* Attackers may attempt to exhaust edge resources by sending large volumes of invalid authentication material. Implementations SHOULD use caching, batching, rate limiting, and hardware acceleration.

* *Replay Attacks:* Authentication proofs MUST include freshness protection such as timestamps, sequence values, nonces, or short-lived authorization tokens.

* *Key Compromise:* The security of an Identity Prefix depends on associated keying material. Deployments MUST define revocation, rotation, and recovery procedures.

* *Locator Poisoning:* Identity-to-locator resolution systems are security-critical and MUST protect against tampering, stale data, and unauthorized updates.

* *Extension Header Abuse:* Malicious packets may use excessive or malformed extension chains to stress parser behavior. Implementations SHOULD enforce total extension length and chain depth limits.

* *Privacy Risk:* Stable identifiers can increase linkability across networks and sessions. Implementations SHOULD support privacy-preserving modes, delegated service identities, or controlled identifier rotation where operationally appropriate.

Where digital signatures are used, Ed25519 or a cryptographically comparable algorithm is RECOMMENDED. Algorithm agility MUST be preserved.

17. Privacy Considerations

Because INUID separates identity from location and allows identity continuity across movement events, it can introduce stronger cross-session correlation than conventional ephemeral addressing. Deployments that require privacy SHOULD define policies for temporary identities, rotating tags, delegated service identities, and minimized observability of stable identity values.

Telemetry and logging systems SHOULD avoid collecting more identity-linked data than necessary for security and operations.

18. IANA Considerations

This document requests the following IANA actions:

1. Allocate IP Version Number 9 for INUID.

2. Create an "INUID Next Header Types" registry.

3. Allocate a Next Header value for the Authentication Extension Header.

4. Allocate a Next Header value for the Mobility Update Extension Header.

5. Allocate a Next Header value for the Telemetry Extension Header.

6. Allocate a Next Header value for the Legacy Translation Context Header.

Future registry assignments SHOULD require Standards Action or IESG Approval.

19. Example Deployment Scenarios

19.1. Data Center Service Mobility

A service retains the same Identity Prefix while migrating between racks or clusters. Only its Locator changes. Existing sessions remain valid because upper layers bind to the stable service identity rather than the old attachment point.

19.2. Enterprise Multihoming

An enterprise advertises multiple locators for a common service identity space across different upstream providers. Traffic can fail over between providers without endpoint renumbering.

19.3. Edge Anti-Spoofing

A residential or enterprise access router validates endpoint identities during attachment. Compromised hosts may still originate traffic, but they cannot forge arbitrary source identities outside their delegated authorization domain.

20. Future Work

Future documents are expected to define the precise wire format of the Authentication Extension Header, standardized locator-resolution procedures, mobility update message formats, an INUID control-message framework, and operational profiles for constrained environments and backbone-class deployments.

21. Conclusion

INUID provides a network-layer framework that separates stable endpoint identity from topological reachability while preserving efficient routing and enabling strong edge-based source validation. By combining hierarchical locators, cryptographic identity, and structured extensibility, INUID offers a foundation for more secure, mobile, and scalable internetworking.

22. Normative References

[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>.

Appendix A. Acknowledgements

The author acknowledges the broader networking research community for foundational work on identifier-locator split architectures, anti-spoofing systems, secure routing, mobility support, and high-performance packet forwarding design.

Author’s Address

Ziad T.
INUID Research Group
Email: contact@inuidresearch.qzz.io
URI: https://inuidresearch.qzz.io