Internet-Draft
Intended status: Informational
Expires: November 2026

M. Norton
Independent

May 2026

# SDLP RFC 1: DigitalID Specification
## draft-norton-sdlp-identity-00

M. Norton
Individual Submission
Email: mark433norton@gmail.com



## Abstract

This document defines the DigitalID specification for the Secured Digital Lifecycle Protocol (SDLP). The DigitalID is the foundational identity construct for all SDLP objects, providing deterministic uniqueness, lineage preservation, collision elimination, and tamper-evident integrity. This document describes the DigitalID structure, assignment rules, lineage model, comparison rules, and integrity requirements.

## Status of This Memo



## Copyright Notice

## 1. Introduction

The DigitalID is the foundational identity construct of the Secured Digital Lifecycle Protocol (SDLP). Every SDLP object, instance, and descendant MUST possess a DigitalID that uniquely identifies the object, encodes its origin and lineage, binds the object to its lifecycle, prevents identity collisions, and enables tamper-evident verification.

This document defines the DigitalID structure, generation rules, lineage model, comparison rules, and integrity requirements.

## 2. Design Goals

The DigitalID is designed to satisfy the following goals:

* Deterministic uniqueness: no two SDLP objects may share the same DigitalID.
* Lineage preservation: all descendant objects MUST encode their ancestry.
* Collision elimination: identity collisions MUST be structurally impossible.
* Tamper resistance: unauthorized modification MUST be detectable.
* Canonical representation: all DigitalIDs MUST be comparable in a byte-exact manner.
* Protocol independence: the DigitalID MUST remain valid across transports, storage systems, and implementations.

## 3. DigitalID Structure

A DigitalID is a structured, hierarchical identifier composed of the following components:

```
DistributorID.CustomerID.ProductID.DownloadID.Lineage.Timestamp
```

Each component is defined as follows:

* DistributorID: the entity that originates or distributes the digital good.
* CustomerID: the entity receiving or activating the digital good.

* ProductID: the unique identifier of the product.

* DownloadID: the unique identifier of the acquisition event.

* Lineage: a dot-separated sequence representing ancestry (e.g., "1", "1.2", "1.2.1").

* Timestamp: the creation or transformation time in canonical form.

### 3.1 Canonical Form

A DigitalID MUST be represented as a UTF-8 string with dot separators. No whitespace, padding, or alternative encodings are permitted.

### 3.2 Comparison Rules

Two DigitalIDs are equal if and only if their canonical UTF-8 byte sequences match exactly. Implementations MUST NOT perform case-insensitive comparison, Unicode normalization, or whitespace trimming.

## 4. Identity Assignment

### 4.1 Origin Identity

When an SDLP object is first created, the Identity Authority (IA) assigns:

* DistributorID
* CustomerID
* ProductID
* DownloadID
* Timestamp
* Lineage = "1"

This DigitalID becomes the root identity of the object.

### 4.2 Descendant Identity

Any duplication, transformation, export, or materialization event MUST produce a descendant DigitalID.

The lineage component is extended as follows:

* First child: 1.1
* Second child: 1.2
* Child of child: 1.2.1

This ensures that no two objects ever share the same DigitalID and that all objects maintain a genealogical chain.

## 5. Collision Model

### 5.1 Collision Prevention

Because DigitalIDs encode origin, acquisition event, lineage, and timestamp, it is impossible for two independently created objects to share the same DigitalID.

### 5.2 Collision Absorption

If an implementation attempts to create an object with an existing DigitalID, the system MUST treat the event as a duplication, generate a descendant DigitalID, extend the lineage component, and preserve

the original identity.

## 5.3 Collision Resolution

Because collisions cannot occur under correct operation, the only collision scenario is tampering (see Section 7).

## 6. Security Properties

The DigitalID MUST provide:

* Immutability: once assigned, the root identity cannot change.
* Lineage integrity: ancestry cannot be removed or rewritten.
* Tamper evidence: unauthorized edits MUST be detectable.
* Non-repudiation: the origin of an object MUST be provable.

These properties are enforced through the SDLP bitdump seal.

## 7. Integrity and Tamper Resistance

### 7.1 Canonical Bitdump

Every SDLP object MUST include a canonical bitdump, which is a byte-exact serialization of the DigitalID, lineage chain, lifecycle state, metadata, and payload (or its hash). This bitdump MUST be stable across implementations.

### 7.2 Seal

A seal is a cryptographic integrity mechanism applied to the canonical bitdump. The seal MUST bind the DigitalID and lineage to the object, prevent unauthorized modification, and allow verification by any SDLP-compliant system.

### 7.3 Tamper Detection

Any modification to DigitalID, lineage, lifecycle state, metadata, or payload that is not performed through a valid SDLP transition MUST cause seal verification to fail.

### 7.4 Tamper Response

If a seal fails verification, the object MUST be rejected, the DigitalID MUST NOT be trusted, the lineage MUST be considered invalid, and the event MUST be logged as a security violation.

### 7.5 Manual Editing

Manual editing of DigitalID or lineage fields is considered tampering. Because the attacker cannot recompute a valid seal without IA authorization, such edits are always detectable.

## 8. IANA Considerations

This document makes no requests of IANA.

## 9. Security Considerations

DigitalID tampering is always detectable. Lineage cannot be forged. Collisions cannot occur. Unauthorized identity modification is impossible without breaking the seal.

## 10. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", RFC 8174, May 2017.

[RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.

Author’s Address

M. Norton
El Mirage, Arizona
United States
Email: mark433norton@gmail.com