Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)

draft-ietf-lisp-8111bis-00

L. Iannone (Ed.), L. Jakab (Ed.)

IETF 124 – Montreal

Since IETF 122: Adopted by WG

Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)
draft-ietf-lisp-8111bis-00

Status | IESG evaluation record | IESG writeups | Email expansions | History

Versions:
00

draft-ietf-lisp-8111bis-00.txt
September 2025
Document adopted and renamed

Some small author adjustment & Security improvement

draft-saucez-lisp-8111bis-01.txt

```
LISP Working Group                                          D. Saucez, Ed.
Internet-Draft                                                       Inria
Obsoletes: 8111 (if approved)                               L. Iannone, Ed.
Updates: 9301 (if approved)                                         Huawei
Intended status: Standards Track                            3 March 2025
Expires: 4 September 2025

   Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)
                    draft-saucez-lisp-8111bis-01
```

Sig-Algorithm: The identifier of the cryptographic algorithm used for the signature. Sig-Algorithm values defined in this specification are listed in Table 1. Implementations conforming to this specification MUST implement at least RSA-SHA256 for DDT signing. Sig-Algorithm type 1 (RSA-SHA1) is deprecated and SHOULD NOT be used.

Sig-Algorithm	Name	Reference	Notes
1	RSA-SHA1	[RFC8017]	DEPRECATED
2	RSA-SHA256	[RFC8017]	MANDATORY

draft-ietf-lisp-8111bis-00.txt

```
LISP Working Group                                          L. Iannone, Ed.
Internet-Draft                                                       Huawei
Obsoletes: 8111 (if approved)                                   L. Jakab, Ed.
Updates: 9301 (if approved)                                         Cisco
Intended status: Standards Track                            3 September 2025
Expires: 7 March 2026

   Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)
                    draft-ietf-lisp-8111bis-00
```

Sig-Algorithm: The identifier of the cryptographic algorithm used for the signature. Sig-Algorithm values defined in this specification are listed in Table 1. Implementations conforming to this specification MUST implement at least RSA-SHA256 for DDT signing. Sig-Algorithm type 1 (RSA-SHA1) is deprecated and SHOULD NOT be used.

Sig-Algorithm	Name	Reference	Notes
1	RSA-SHA1	[RFC8017]	DEPRECATED
2	RSA-SHA256	[RFC6234]	MANDATORY
3	RSA-SHA512	[RFC6234]	RECOMMENDED

- Welcome back Lorand ;-)
- A lot of typos correction
- Added stronger security

Open Issues: Inconsistent IID and DBID usefulness

4.1. XEID-Prefixes

A DDT database is indexed by Extended EID-Prefixes (XEID-Prefixes). An XEID-Prefix is a LISP EID-Prefix that includes additional data to uniquely identify the address space associated with the prefix. An XEID-Prefix is composed of four binary-encoded concatenated attributes:

* DBID (16 bits),
* Instance ID (24 bits),
* AFI (16 bits),
* and EID-Prefix (variable, according to the AFI value).

[lisp] Re: IID lengnth

Luigi Iannone <ggx@gigix.net> | Wed, 26 March 2025 13:36 UTC | Show header

That is not enough.

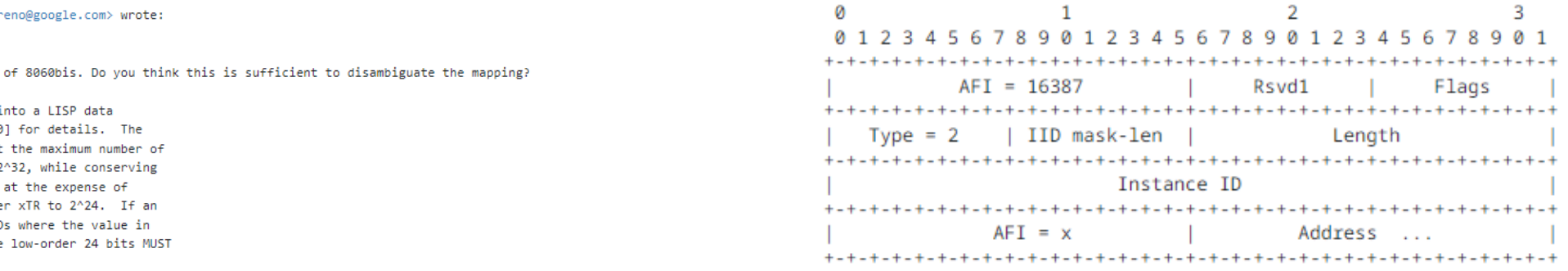
L.

> On 26 Mar 2025, at 14:32, Dino Farinacci <farinacci@gmail.com> wrote:
>
> Yes - since Alvaro is updating 8060bis he can add text. Simply say low order 24 bits of the 32 bit IID is used as the 24 bits IID in the data plane.
>
> Dino
>
>> On Mar 26, 2025, at 6:27 AM, Victor Moreno <vimoreno@google.com> wrote:
>>
>>
>> I pasted below the text currently in section 4.1 of 8060bis. Do you think this is sufficient to disambiguate the mapping?
>>
>> Instance ID: the low-order 24 bits that can go into a LISP data
>> header when the I bit is set. See [RFC9300] for details. The
>> reason for the length difference is so that the maximum number of
>> instances supported per mapping system is 2^32, while conserving
>> space in the LISP data header. This comes at the expense of
>> limiting the maximum number of instances per xTR to 2^24. If an
>> xTR is configured with multiple Instance IDs where the value in
>> the high-order 8 bits is the same, then the low-order 24 bits MUST
>> be unique

- Settled for an XEID being the concatenation of
  - 16 bits of the DBID
  - 24 bits of the IID
  - 16 bits of the AFI
  - And the prefix
- Implementations encoding the IID on 32 bits internally are not affected since "on the wire" in the various messages the IID is never encoded as 32 bits
- Database Identifier (DBID) "allows the definition of multiple databases" & "implementation compliant with this document must set this field to 0" this value is more a DB type and is never seen on the wire.

- **Do we need to keep it??**

Message: https://mailarchive.ietf.org/arch/msg/lisp/5KkbD6kmPAcF99ksGgkyTfK3S0E/



Next Steps

- IID Encoding (All)
- DBID (All)
- Error handling (Luigi)
- Deployment experience (Lorand)

THANKS!