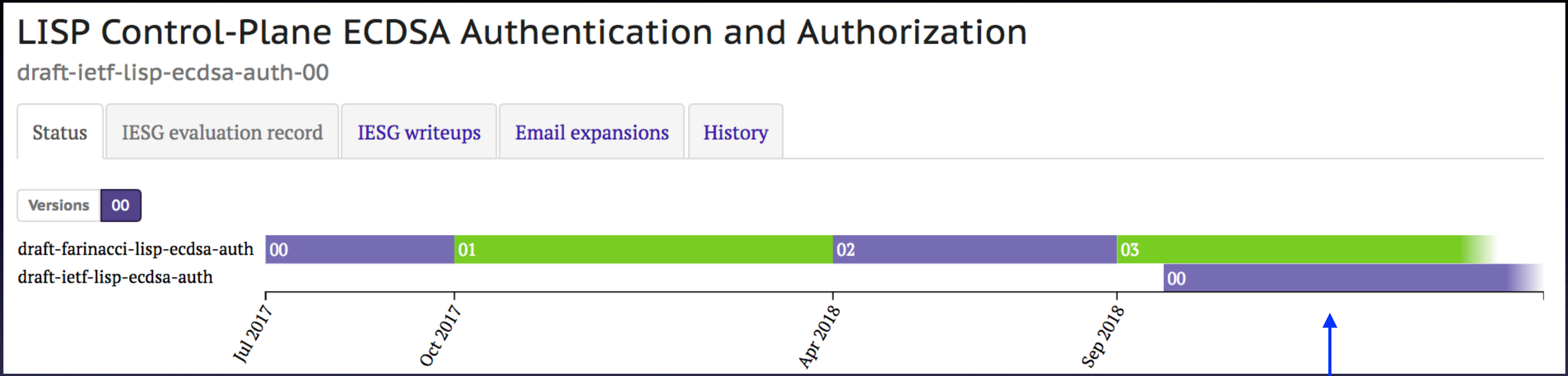
LISP Digital Signatures

draft-ietf-lisp-ecdsa-auth-00

IETF LISP WG Bangkok
November 2018

Dino Farinacci & Erik Nordmark

Document Status



LISP Control-Plane ECDSA Authentication and Authorization

draft-ietf-lisp-ecdsa-auth-00

Initial individual

Update pre-Singapore

Update post-London

Final Individual post-Montreal

Initial WG Sep 2018

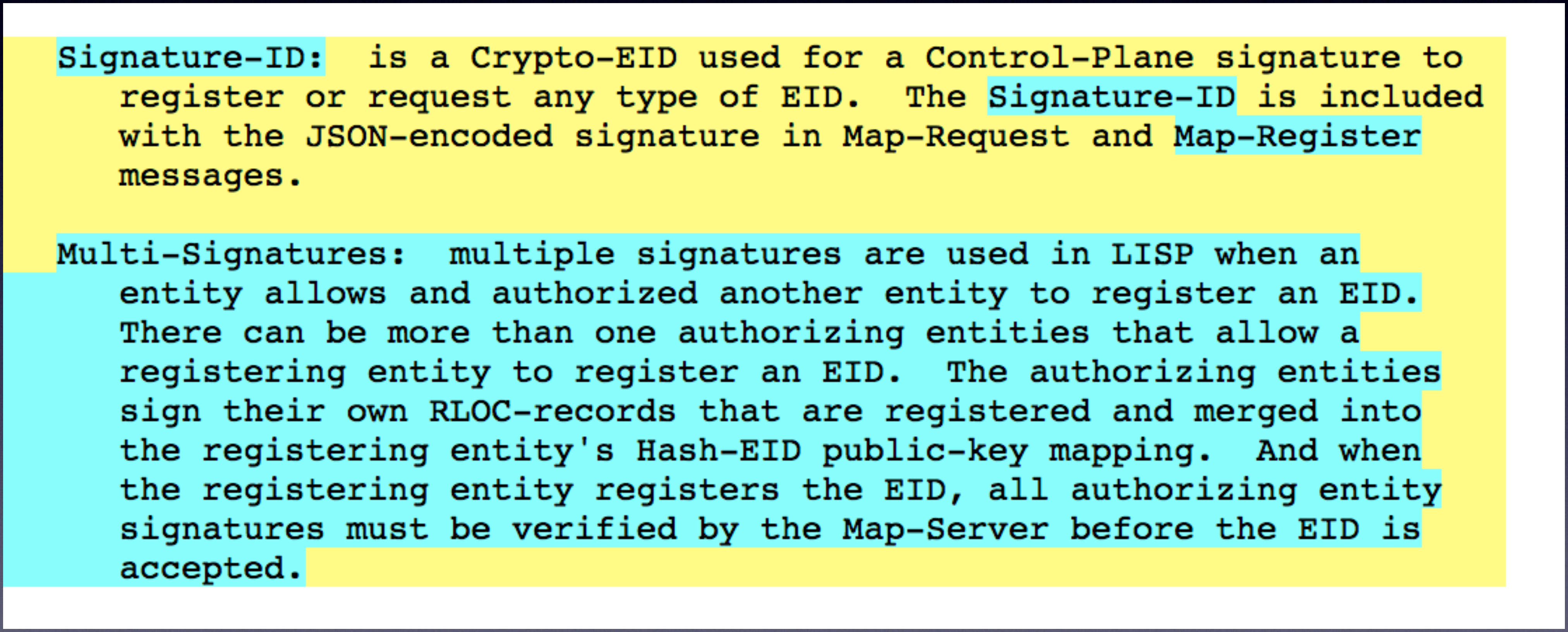
Draft Overview

- Authenticate & authorize xTRs using the mapping system
- How to sign Map-Registers
- How to sign Map-Requests
- How to store public-keys in mapping system
- Introduces Crypto-EIDs
- Introduces Signature-IDs (previously called Signature-EIDs)

Benefits

- Strong Elliptic Curve Cryptography using DSA
- Can verify and invalidate a single xTR
- Can use the signature-ID for registering any EID type
- Can use public-key for encrypting results sent back to xTR
- Provides identity privacy - multiple key-pairs can be used

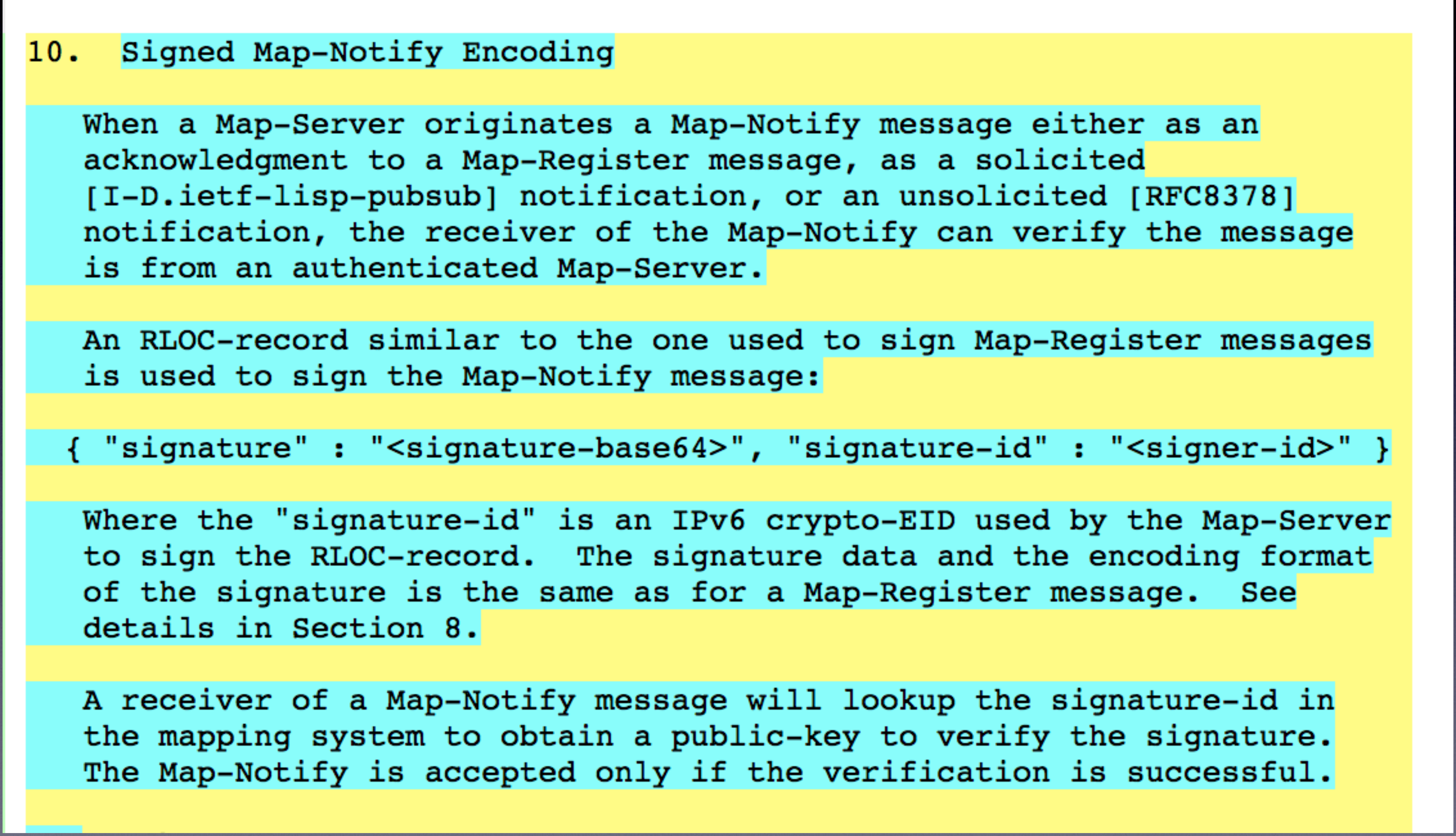
Contents in -03/00



Signature-ID: is a Crypto-EID used for a Control-Plane signature to register or request any type of EID. The Signature-ID is included with the JSON-encoded signature in Map-Request and Map-Register messages.

Multi-Signatures: multiple signatures are used in LISP when an entity allows and authorized another entity to register an EID. There can be more than one authorizing entities that allow a registering entity to register an EID. The authorizing entities sign their own RLOC-records that are registered and merged into the registering entity's Hash-EID public-key mapping. And when the registering entity registers the EID, all authorizing entity signatures must be verified by the Map-Server before the EID is accepted.

Contents in -03/00



10. Signed Map-Notify Encoding

When a Map-Server originates a Map-Notify message either as an acknowledgment to a Map-Register message, as a solicited [I-D.ietf-lisp-pubsub] notification, or an unsolicited [RFC8378] notification, the receiver of the Map-Notify can verify the message is from an authenticated Map-Server.

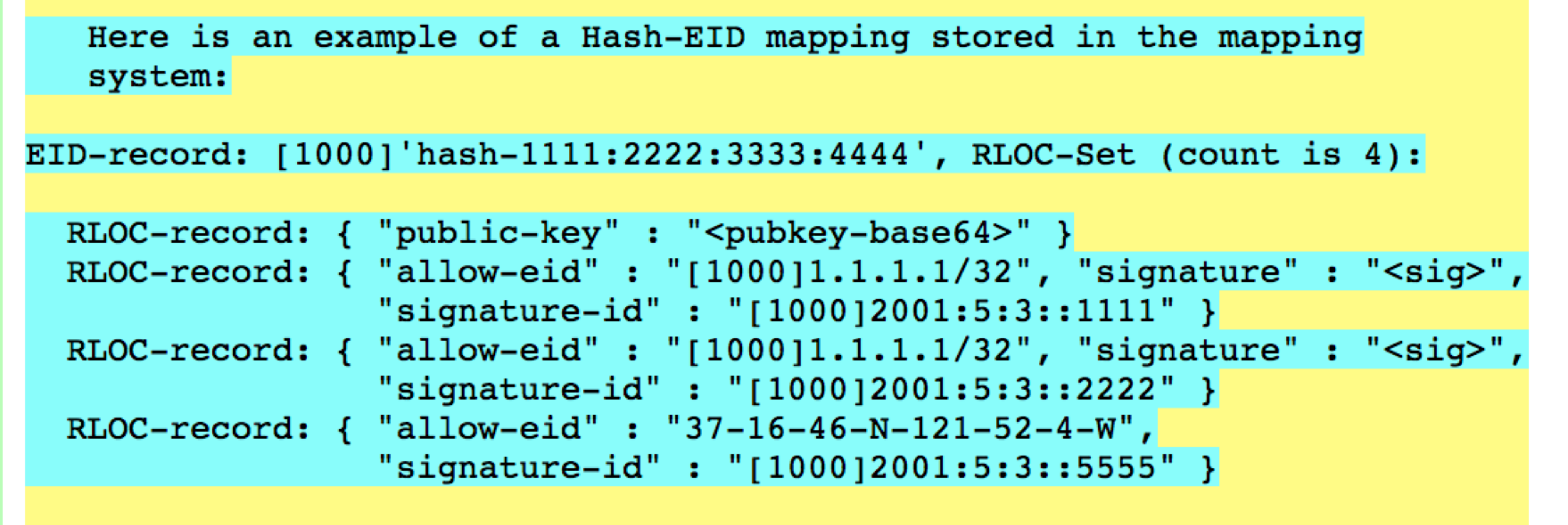
An RLOC-record similar to the one used to sign Map-Register messages is used to sign the Map-Notify message:

```
{ "signature" : "<signature-base64>", "signature-id" : "<signer-id>" }
```

Where the "signature-id" is an IPv6 crypto-EID used by the Map-Server to sign the RLOC-record. The signature data and the encoding format of the signature is the same as for a Map-Register message. See details in Section 8.

A receiver of a Map-Notify message will lookup the signature-id in the mapping system to obtain a public-key to verify the signature. The Map-Notify is accepted only if the verification is successful.

Contents in -03/00



```
   Here is an example of a Hash-EID mapping stored in the mapping
   system:

EID-record: [1000]'hash-1111:2222:3333:4444', RLOC-Set (count is 4):

  RLOC-record: { "public-key" : "<pubkey-base64>" }
  RLOC-record: { "allow-eid" : "[1000]1.1.1.1/32", "signature" : "<sig>",
                 "signature-id" : "[1000]2001:5:3::1111" }
  RLOC-record: { "allow-eid" : "[1000]1.1.1.1/32", "signature" : "<sig>",
                 "signature-id" : "[1000]2001:5:3::2222" }
  RLOC-record: { "allow-eid" : "37-16-46-N-121-52-4-W",
                 "signature-id" : "[1000]2001:5:3::5555" }
```

Possible Todo List

- Spec how RLOC-probe Map-Requests signatures can be verified by **ETRs** and RLOC-probe Map-Replies by **ITRs**
- Consider encrypting Map-Registers from **ETR** to **Map-Server** using public-key of Map-Server (but can use shared-key right now)
- Consider encrypting Map-Requests from **ITR** to **Map-Resolver** using public-key of Map-Resolver (LISP-DDT takes it from here to Map-Server)
- Consider encrypting Map-Replies from **ETR/MS** to **ITR** using public-key of ITR
- Consider encrypting Map-Notifies from **MS** to **ITR** using public-key of ITR for **PubSub**

Questions?