Network Working Group
Internet-Draft
Intended status: Experimental
Expires: September 14, 2012

V. Fuller
D. Lewis
V. Ermagan
cisco Systems
March 13, 2012

LISP Delegated Database Tree
draft-fuller-lisp-ddt-01.txt

Abstract

This draft describes the LISP Delegated Database Tree (LISP-DDT), a hierarchical, distributed database which embodies the delegation of authority to provide mappings from LISP Endpoint Identifiers (EIDs) to Routing Locators (RLOCs). It is a statically-defined distribution of the EID namespace among a set of LISP-speaking servers, called DDT nodes. Each DDT node is configured as "authoritative" for one or more EID-prefixes, along with the set of RLOCs for Map-Servers or "child" DDT nodes to which more-specific EID-prefixes are delegated.

Table of Contents

1. Introduction

[LISP] specifies an architecture and mechanism for replacing the addresses currently used by IP with two separate name spaces: relatively static Endpoint Identifiers (EIDs), used end-to-end for terminating transport-layer associations, and Routing Locators (RLOCs), which are more dynamic, are bound to topological location, and are used for routing and forwarding through the Internet infrastructure.

LISP offers a general-purpose mechanism for mapping between EIDs and RLOCs. In organizing a database of EID to RLOC mappings, this specification extends the definition of the EID numbering space by logically prepending and appending several fields for purposes of defining the database index key: Key-ID (16 bits), Instance Identifier (IID, 32-bits), Address Family Identifier (16 bits), and EID-prefix (variable, according to AFI value). The resulting concatenation of these fields is termed an "Extended EID prefix" or XEID-prefix.

The term "Extended EID" (XEID) is also used for an individual LISP EID that is further qualified through the use of an Instance ID. See [LCAF] for further discussion of the use of Instance IDs.

The Key-ID is provided for possible use in case a need evolves for another, higher level in the hierarchy, to allow the creation of multiple, separate database trees.

LISP-DDT is a hierarchical distributed database which embodies the delegation of authority to provide mappings, i.e. its internal structure mirrors the hierarchical delegation of address space. It also provides delegation information to Map-Resolvers, which use the information to locate EID-to-RLOC mappings. A Map-Resolver which needs to locate a given mapping will follow a path through the tree-structured database, contacting, one after another, the DDT nodes along that path until it reaches the leaf DDT node(s) authoritative for the mapping it is seeking.

LISP-DDT defines a new device type, the DDT node, that is configured as authoritative for one or more XEID-prefixes. It also is configured with the set of more-specific sub-prefixes that are further delegated to other DDT nodes. To delegate a sub-prefix, the "parent" DDT node is configured with the RLOCs of each child DDT node that is authoritative for the sub-prefix. Each RLOC either points to a Map-Server (sometimes termed a "terminal DDT node") to which an Egress Tunnel Routers (ETRs) registers that sub-prefix or points to another DDT node in the database tree that further delegates the sub-prefix. See [LISP-MS] for a description of the functionality of the

Map-Server and Map-Resolver. Note that the target of a delegation must always be an RLOC (not an EID) to avoid any circular dependency.

To provide a mechanism for traversing the database tree, LISP-DDT defines a new LISP message type, the Map-Referral, which is returned to the sender of a Map-Request when the receiving DDT node can refer the sender to another DDT node that has more detailed information. See Appendix B for the definition of the Map-Referral message.

A DDT client uses LISP-DDT to find an EID-to-RLOC mapping by first sending a Map-Request to the RLOC of a DDT node. The initial choice of DDT node is configured on the client. If the receiving DDT node is also a Map-Server that is responsible for the XEID queried, the Map-Request is handled as described in [LISP-MS], with the DDT Map-Server also returning a Map-Referral message with the "done" flag set to the Map-Request sender. Otherwise, the DDT node answers the Map-Request with a Map-Referral; the DDT client then re-sends its DDT Map-Request to one of the RLOCs listed in the Map-Referral. This iterative process of sending requests and following referrals continues until the client receives a Map-Referral with the "done" flag set. This is an indication that the terminal DDT Map-Server has either answered the Map-Request (if offering proxy service, as described in [LISP-MS]) or has forwarded it to the correct ETR which will answer it. Conceptually, this is similar to the way that a client of the Domain Name System (DNS) follows referrals (DNS responses that contain only NS records) from a series of DNS servers until it finds an answer.

2. Definition of Terms

Extended EID (XEID): a LISP EID, optionally extended with a non-zero Instance ID (IID) if the EID is intended for use in a context where it may not be a unique value, such as on a Virtual Private Network where [RFC1918] address space is used. See "Using Virtualization and Segmentation with LISP" in [LISP] for more discussion of Instance IDs.

XEID-prefix: a LISP EID-prefix with 16-bit LISP-DDT Key-ID (provided to allow the definition of multiple databases; currently always zero in this version of DDT, with other values reserved for future use), 32-bit IID and 16-bit AFI prepended. An XEID-prefix is used as a key index into the database.

DDT node: a network infrastructure component responsible for specific XEID-prefix and for delegation of more-specific sub-prefixes to other DDT nodes.

DDT client: a network infrastructure component that sends Map-Request messages and implements the iterative following of Map-Referral results. Typically, a DDT client will be a Map-Resolver but it is also possible for an ITR to implement DDT client functionality.

DDT Map-Server: a DDT node that also implements Map Server functionality (forwarding Map-Requests and/or returning Map-Replies if offering proxy-mode service) for a subset of its delegated prefixes.

DDT Map-Resolver: a network infrastructure element that accepts a Map-Request, adds the XEID to its lookup queue, then queries one or more DDT nodes for the requested EID, following returned referrals until it receives one with the "done" flag. This indicates that the Map-Request has been sent to a Map-Server that will forward it to an ETR that, in turn, will provide a Map-Reply to the original sender. A DDT Map-Resolver maintains both a cache of Map-Referral message results containing RLOCs for DDT nodes responsible for XEID-prefixes of interest (termed the "referral cache") plus a lookup queue of XEIDs that are being resolved through iterative querying of DDT nodes.

Encapsulated Map-Request: a LISP Map-Request carried within an Encapsulated Control Message, which has an additional LISP header prepended. Sent to UDP destination port 4342. The "outer" addresses are globally-routable IP addresses, also known as RLOCs. Used by an ITR when sending to a Map-Resolver and by a Map-Server when forwarding a Map-Request to an ETR as documented in

[LISP-MS].

DDT Map-Request: an Encapsulated Map-Request sent by a DDT client to a DDT node. The "DDT-originated" flag is set in the encapsulation header indicating that the DDT node should return Map-Referral messages if the Map-Request EID matches a delegated XEID-prefix known to the DDT node. Section 7.1 describes how DDT Map-Requests are sent.

Authoritative XEID-prefix: an XEID-prefix delegated to a DDT node and for which the DDT node may provide further delegations of more-specific sub-prefixes.

Map-Referral: a LISP message sent by a DDT node when it receives a DDT Map-Request for an XEID that matches a configured XEID-prefix delegation. The Map-Referral message includes a "referral", a set of RLOCs for DDT nodes that have more information about the sub-prefix; a DDT client "follows the referral" by sending another DDT Map-Request to one of those RLOCs to obtain either an answer or another referral to DDT nodes responsible for a more-specific XEID-prefix. See Section 5 and Section 7.2 for details on the sending and processing of Map-Referral messages.

negative Map-Referral: a LISP message sent by a DDT node when it receives a DDT Map-Request for an EID that matches a configured authoritative XEID-prefix but for which no delegation (or registration if the DDT node is also a Map-Server) is configured.

For definitions of other terms, notably Map-Request, Map-Reply, Ingress Tunnel Router (ITR), Egress Tunnel Router (ETR), Map Server, and Map-Resolver, please consult the LISP specification [LISP] and the LISP Mapping Service specification [LISP-MS].

3. EID-prefix tree structure and instance IDs

LISP-DDT defines a tree structure that is indexed by a binary encoding of five fields, in order of significance: Key-ID (16 bits), Instance Identifier (IID, 32 bits), Address Family Identifier (AFI, 16 bits), and EID-prefix (variable, according to AFI value). The fields are concatenated, with the most significant fields as listed above. The index into this structure is also referred to as an Extended EID-prefix (XEID-prefix).

It is important to note that LISP-DDT does not store actual EID-to-RLOC mappings; it is, rather, a distributed index that can be used to find the devices (Map-Servers and their registered EIDs) that can be queried with LISP to obtain those mappings. Changes to EID-to-RLOC mappings are made on the ETRs which define them, not to any DDT node configuration. DDT node configuration changes are only required when branches of the database hierarchy are added, removed, or modified.

4. Configuring XEID-prefix delegation

Every DDT node is configured with one or more XEID-prefixes for which it is authoritative along with a list of delegations of XEID-prefixes that are known to other DDT nodes. A DDT node is required to maintain a list of delegations for all sub- prefixes of its authoritative XEID-prefixes but also may list "hints", which are prefixes that it knows about that belong to its parents, to the root, or to any other point in the XEID-prefix hierarchy. A delegation (or hint) consists of an XEID-prefix, a set of RLOCs for DDT nodes that have more detailed knowledge of the XEID-prefix, and acompanying security information. Those RLOCs are returned in Map-Referral messages when the DDT node receives a DDT Map-Request with an xEID that matches a delegation. A DDT Map-Server will also have a set of sub-prefixes for which it accepts ETR mapping registrations and for which it will forward (or answer, if it implements proxy mode) Map-Requests. For details of security infomation in Map-Referrals see Section 9.

4.1. Example DDT node configuration

The following is an example of parent and child DDT nodes, where the parent has all of 10.0.0.0/8 and delegates two sub-prefixes, 10.0.0.0/12 and 10.0.16.0/12 to two child DDT nodes. All of these prefixes are within the DDT sub-tree Key-ID=0, IID=223, and AFI=1 (IPv4).

```
lisp ddt authoritative-prefix instance-id 223 10.0.0.0/8
lisp ddt child 192.168.1.100 instance-id 223 eid-prefix 10.0.0.0/12
lisp ddt child 192.168.1.200 instance-id 223 eid-prefix 10.16.0.0/12
```

This defines delegation of the EID-prefix 10.0.0.0/12 to a DDT Map Server with RLOC 192.168.1.100 and delegation of the EID-prefix 10.16.0.0/12 to a DDT Map-Server with RLOC 192.168.1.200.

The child DDT Map-Server for 10.16.0.0/12 is further configured to allow ETRs to register the sub-prefixes 10.18.0.0/16 and 10.17.0.0/16:

```
lisp ddt authoritative-prefix instance-id 223 eid-prefix 10.16.0.0/12
lisp site site-1
  eid-prefix 10.18.0.0/16 instance-id 223
lisp site site-2
  eid-prefix 10.17.0.0/16 instance-id 223
```

4.2. The root DDT node

The root DDT node is the logical "top" of the database hierarchy: Key-ID=0, EID=0, AFI=0, EID-prefix=0/0. A DDT Map-Request that matches no configured XEID-prefix will be referred to the root node. The root node in a particular instantiation of LISP-DDT must therefore be configured with delegations for at least all defined IIDs and AFIs.

To aid in defining a "sub-root" DDT node that is responsible for all EID-prefixes within multiple IIDs (say, for using LISP to create virtual networks that use overlapping address space), it may be useful to implement configuration language that allows for a range of IIDs to be delegated together. Additional configuration shorthand for delegating of a range of IIDs (and all of the EIDs under them) may also be helpful.

5. DDT node operation - sending referrals

When a DDT node receives a DDT Map-Request, it compares the requested XEID against its list of XEID-prefix delegations and its list of authoritative XEID-prefixes and acts as follows:

5.1. Match of a delegated prefix (or sub-prefix)

If the requested XEID matches one of the DDT node's delegated prefixes, then a Map-Referral message is returned with the matching more-specific XEID-prefix and the set of RLOCs for the referral target DDT nodes including associated security information (see Section 9 for details on security).

Note that a matched delegation does not have to be for a sub-prefix of an authoritative prefix; in addition to being configured to delegate sub-prefixes of an authoritative prefix, a DDT node may also be configured with other XEID-prefixes for which it can provide referrals to DDT nodes anywhere in the database hierarchy. This capability to define "shortcut hints" is never required to be configured but may be a useful heuristic for reducing the number of iterations needed to find an EID, particular for private network deployments.

5.2. Missing delegation from an authoritative prefix

If the requested XEID did not match a configured delegation but does match an authoritative XEID-prefix, then the DDT node returns a negative Map-Referral that includes the least-specific XEID-prefix that does not match any of the DDT node's authoritative XEID-prefixes, including associated security information. This indicates that the XEID is not a LISP destination.

If the requested XEID did not match either a configured delegation or an authoritative XEID-prefix, then the request is dropped. This should only happen if either a DDT Map-Resolver or DDT Map-Server is misconfigured. Logging an error message may be a good idea to assist in detecting and resolving such configuration problems.

6. DDT Map-Server operation

When a DDT Map-Server receives a DDT Map-Request, its operation is similar to that of a DDT node with one exception: if the requested XEID matches a registered XEID-prefix, then the Map-Request is forwarded to one of the destination ETR RLOCs (or the Map-Server sends a Map-Reply, if it is providing proxy service) and a Map-Referral with the MS-ACK action is returned to the sender of the DDT Map-Request.

7. DDT Map-Resolver operation

Just as any other Map-Resolver, a DDT Map-Resolver accepts Map-Requests from its clients (typically, ITRs) and ensures that those Map-Requests are forwarded to the correct ETR, which generates Map-Replies. Unlike a Map-Resolver that uses the ALT mapping system [LISP-ALT], however, a DDT Map-Resolver needs to maintain more state as it uses an iterative process of following referrals to find the correct ETR to answer a Map-Request.

7.1. Queuing, Sending, and Retransmitting DDT Map-Requests

When a DDT Map-Resolver receives an encapsulated Map-Request, it first performs a longest-match search for the XEID in its referral cache. If nothing is found or if a negative cache entry is found, then the destination is not in the database; a negative Map-Reply is returned and no further processing is done by the DDT Map-Resolver.

Next, the DDT Map-Resolver creates a lookup queue entry for the XEID and saves the original Map-Request along with other information, such as the longest XEID-prefix matched so far, needed for tracking progress through the iterative referral process. The Map-Resolver then creates a DDT Map-Request (which is an encapsulated Map-Request with the "DDT-originated" flag set in the message header) for the XEID but without any authentication data that may have been included in the original Map-Request. It sends the DDT Map-Request to one of the RLOCs in the chosen referral cache entry. The referral cache is initially populated with one or more statically-configured entries; additional entries are added when referrals are followed, as described below. A DDT Map-Resolver is not absolutely required to cache referrals but it not doing so will significantly increase latency and cause lookup delays.

Note that in normal use on the public Internet, the statically-configured initial referral cache for a DDT Map-Resolver should include a "default" entry with RLOCs for one or more DDT nodes that can reach the DDT root node. If a Map-Resolver does not have such configuration, it will return a Negative Map-Reply if it receives a query for an EID outside the subset of the mapping database known to it. While this may be desirable on private network deployments or during early transition to LISP when few sites are using it, this behavior is not appropriate when LISP is in general use on the Internet.

7.2. Receiving and following referrals

After sending a DDT Map-Request, a DDT Map-Resolver can expect one of the following to occur:

- No response. The DDT Map-Resolver retransmits the request, choosing a different RLOC from the referral cache entry if one is available. If the maximum number of retransmissions has occurred, then the lookup queue entry is dequeued and a negative Map-Reply is returned to the original Map-Request sender.

- A Map-Referral message. This indicates that the replying DDT node or DDT Map-Server doesn't know the ETRs for the specific XEID but does know another DDT node or DDT Map-Server that has information about a matching XEID-prefix. The Map-Referral message contains a "map-record" with additional information that is used by a DDT Map-Resolver to "follow" the referral. The subsections below describe how a DDT Map-Resolver uses the fields in the Map-Referral message to detemine the next step in processing a lookup queue entry. See Appendix B for a detailed description of all Map-Referral message fields.

7.2.1. Referral Set

The set of RLOCs for DDT-nodes that are authoritative for the XEID-prefix returned in the Map-Referral message. A DDT Map-Resolver sends subsequent Map-Requests to one or more of these RLOCs as it "follows" a referral.

7.2.2. Referral list incomplete flag

The "Incomplete" flag is set by a DDT Node when it returns a Map-Referral message if the Referral Set is incomplete. A DDT Map-Resolver receiving such a message will need to take appropriate action, specifically not caching the referral. A DDT node must set the "incomplete" flag in the following cases:

- The DDT Map-Server would return Map-Referral with the type of either MS-ACK or ms-not-registerd, but it does not have any configuration about other, "peer" Map-Servers for that also authoritative for the matched XEID-prefix.

- The DDT node returns a Map-Referral with the action of NOT-AUTHORITATIVE.

7.2.3. Action Types

A DDT node sets the "Action" field to indicate to a Map-Resolver what action it should take upon receipt of a Map-Referral message. The defined actions are as follows:

Type 0, NODE-REFERRAL: Follow the referral by saving the prefix in the referral cache and sending a new Map-Request to the first DDT node listed in the Referral Set. A DDT node sends this action code to instruct a DDT Map-Resolver to query a child DDT node.

Type 1, MS-REFERRAL: Follow the referral by saving the prefix in the referral cache and sending a new Map-Request to the first DDT Map-Server listed in the Referral Set. A DDT node sends this action code to instruct a DDT Map-Resolver to query a DDT Map-Server which should have one or more ETRs registered for the matched XEID-prefix.

Type 2, MS-ACK: If the "incomplete" flag is not set, then the referral process is complete; save the prefix in the referral cache and de-queue the original Map-Request. A DDT Map-Server sends an "MS-ACK" response to a DDT Map-Resolver when it forwards a Map-Resolver-originated Map-Request to an ETR.

Type 3, MS-NOT-REGISTERED: This action code indicates that a DDT Map-Server has received a Map-Request for one of its XEID-prefixes but for which it has no ETR registered. If the DDT Map-Resolver has not yet tried all of the DDT Map-Server RLOCs in its referral cache entry, then sends a Map-Request to the next available DDT Map-Server RLOC. If all RLOCs have been tried, then the destination XEID is not registered and is unreachable. The Map-Resolver returns a negative Map-Reply to the original Map-Request sender; this Map-Reply contains the non-registered XEID prefix with TTL value of one minute. It also removes the lookup queue entry.

Type 4, DELEGATION-HOLE: Cache the prefix and return a negative Map-Reply to the original Map-Request sender. The negative Map-Request will indicate the least-specific XEID- prefix matching requested XEID for which no delegations exist; it is sent with a TTL value of 15 minutes.

Type 5, NOT-AUTHORITATIVE: A DDT node returns this action code if it receives a Map-Request for an XEID-request for which it is not authoritative. This can occur if a cached referral has become invalid due to a change in the database hierarchy. If the a DDT Map-Resolver that receives this action code can determine that it is using old cached information, it may choose to delete that cached information and re-try the original Map-Request, starting from its "root" cache entry. If this action code is received in response to a query that was not to cached referral information, then it indicates a serious misconfiguration in the database; the original Map-Request should be removed, unanswered, from the lookup queue.

7.2.4. Handling referral errors

Other states are possible, such as a misconfigured DDT node (acting as a proxy Map-Server, for example) returning a Map-Reply to the DDT Map-Resolver; they should be considered errors and logged as such. It is not clear exactly what else the DDT Map-Resolver should do in such cases; one possibility is to dequeue the lookup queue entry and send a negative Map-Reply to the original Map-Request sender. Alternatively, if a DDT Map-Resolver detects unexpected behavior by a DDT node, it could mark that node as unusable in its referral cache and update the lookup queue entry to try a different DDT node if more than one is listed in the referral cache.

7.2.5. Referral loop detection

With any iterative process, there is always the danger of an iteration loop. To prevent this, a DDT Map-Resolver keeps track of the most recent "referral XEID-prefix" in each lookup queue entry. When it receives a Map-Referral message, it performs the following check for looping:

- For Action Types NODE-REFERRAL and MS-REFERRAL, the new XEID-prefix must be more-specific than the saved prefix; an exact or less-specific match, indicates a referral loop.

- For Action Types MS-ACK, MS-NOT-REGISTERED, or DELEGATION-HOLE, the new XEID-prefix must be an exact or more-specific match of the saved prefix; a less-specific match indicates a referral loop. The exact match is allowed here since these messages indicate that the referral process has completed. Note, though, that the cached RLOCs are NOT updated for an exact match since doing so may lose information needed for preventing loops.

If a loop is detected, then the Map-Resolver handles the request as described in Section 7.2.4. Otherwise, the Map-Resolver saves the most rececent referral XEID-prefix in the lookup queue entry when it follows the referral.

As an extra measure to prevent referral loops, it is probably also wise to limit the total number of referrals for any request to some reasonable number; the exact value of that number will be determined during experimental deployment of LISP-DDT but is bounded by the maximum length of the XEID.

Note that when a Map-Request is originally received and an entry has been added to the lookup queue, the new request has no previous referral XEID-prefix; this means that the first DDT node contacted by a DDT Map-Resolver may provide a referral to anywhere in the DDT

hierarchy. This, in turn, allows a DDT Map-Resolver to use essentially any DDT node RLOCs for its initial cache entries and depend on the initial referral to provide a good starting point for Map-Requests; there is no need to configure the same set of root DDT nodes in all DDT Map-Resolvers.

8. Example message flow

The following describes the message flows among an ITR, a DDT Map Resolver, a number of DDT nodes, a DDT Map-Server, and an ETR. It assumes no security associations between the DDT nodes but does show how [LISP-SEC] can be used between the ITR, Map Resolver, Map-Server, and ETR.

8.1. ITR sends a Map-Request to a DDT Map-Resolver

The first step in using LISP-DDT is the same as for any other Map-Request using the Map-Server interface: an ITR sends an encapsulated Map-Request to one of its configured Map-Resolvers, in this case a DDT Map-Resolver. The outer header source IP address is the ITR and the outer header destination IP address is the DDT Map Resolver. If [LISP-SEC] is in use, then LISP-SEC ECM Authentication Data field is included.

8.2. DDT Map-Resolver receives and processes Map-Request

The DDT Map-Resolver receives and processes the encapsulated Map-Request by stripping the encapsulation header and creating a lookup queue entry for the XEID, saving the resulting, non-encapsulated Map-Request for later retransmission and re-use during the referral process. If [LISP-SEC] information was included in the original, encapsulated Map-Request then it is also saved in the lookup queue entry for later use. The lookup queue entry will be dequeued when the DDT Map-Resolver is finished with the request (see Section 8.8).

Note that if a lookup queue entry already exists for the destination XEID and the requesting ITR (which could happen if an ITR has retransmitted a Map-Request), the Map-Request is replaced to ensure that the ITR-generated nonce and any ECM Authentication Data field are updated.

8.3. DDT Map-Resolver searches referral cache for XEID

Next, the DDT Map-Resolver searches its referral cache for the XEID. If none is found or if a negative cache entry is found, then the XEID does not exist in the database; a negative Map-Reply is returned to the original sender and the lookup queue entry is dequeued.

If the referral cache entry found is for a DDT Map-Server, then the DDT Map-Resolver has found the appropriate terminal node in the DDT hierarchy. It finishes processing the lookup queue entry as described in Section 8.8.

At this point, the referral cache entry must be for a DDT node that

can provide more-specific information for the requested XEID so a DDT Map-Request is created and sent (see below).

8.4. DDT Map-Resolver creates and sends DDT Map-Request

To follow a referral and query the next DDT node, the DDT Map Resolver creates a new DDT Map-Request, an encapsulated Map-Request using one of the RLOCs of the target DDT node as the outer header destination IP address and itself as the outer header source IP address. The "DDT-originated" flag is set in the encapsulation header to inform the target DDT node that it should return referrals. The original Map-Request LISP-SEC information, if any was saved in the lookup queue entry, is NOT included. The original Map-Request destination XEID is used in the new Map-Request while the source is one of the DDT Map-Resolver's RLOCs.

The new "DDT Map-Request" is transmitted to the destination DDT node. If no response is received within a timeout, it is re-transmitted, preferably using a different destination DDT node RLOC. If the maximum number of retransmissions is exceeded, the request is dequeued and a negative Map-Reply is returned to the ITR that sent the original Map-Request.

8.5. DDT node receives and processes DDT Map-Request

The destination DDT node searches its configured delegations and authoritative prefixes for the XEID in the received encapsulated Map-Request. If no match is found, then the DDT Map-Request is silently discarded and, optionally, an error is logged.

If a delegation is found, the DDT node sends a Map-Referral message back to the DDT Map-Resolver with the matched XEID-prefix and the set of RLOCs for DDT nodes that can be used to resolve XEIDs within that prefix.

If no matching delegation was found and the XEID matches one of the DDT node's authoritative prefixes, then the destination is not a LISP XEID (or a configuration error has occurred); the DDT node returns a negative Map-Referral message to the DDT Map-Resolver as described in Section 5.2.

8.6. DDT Map-Resolver processes Map-Referral

When the DDT Map-Resolver receives a Map-Referral from a DDT-node, it first verifies that it has a corresponding lookup queue entry; if none can be found, then the Map-Referral is silently ignored, with optional error logging.

If the received Map-Referral was negative, then the destination XEID is not in the database; a negative Map-Reply is returned to the original Map-Request sender, a negative referral cache entry is created for the returned XEID-prefix (with TTL from the Map-Referral message), and the lookup queue entry is dequeued.

For a non-negative Map-Referral, the lookup queue entry is updated with the new referral XEID-prefix and new DDT-node RLOCs. At this point, it also checks to make sure that a referral loop has not occurred (see Section 7.2.5).

To speed processing of future Map-Requests for the same XEID-prefix, the DDT Map-Resolver adds a new entry (or updates an existing, matching entry) in its referral cache for the XEID-prefix, RLOC set, and TTL value in the Map-Referral message. Finally, processing continues to Section 8.4 to query the new destination DDT-node.

8.7. DDT Map-Server receives Map-Request

At this point, the DDT Map-Resolver has found the DDT Map-Server responsible for the destination XEID-prefix and has sent its Map-Request there. The DDT Map-Server receives the DDT Map-Request, strips the encapsulation header, and searches for the destination XEID in its set of configured XEID-prefixes. If the XEID is found and an ETR has registered for it, then DDT Map-Server returns a Map-Referral to the DDT Map-Resolver indicating (by setting the MS-ACK action) that it has found the terminal DDT node. The Map-Request is forwarded to one of the registered ETRs for further processing (Section 8.10).

8.8. DDT Map-Resolver finished

At this point, the DDT Map-Resolver has finished the referral iteration process. If security processing was requested, the DDT Map Resolver now re-sends the DDT Map-Request to the DDT Map-Server with the LISP-SEC information included in the encapsulation header. The DDT Map-Resolver dequeues the lookup queue entry for the XEID and cleans-up any other saved state.

8.9. DDT Map-Server receives LISP-SEC-enabled Map-Request

When the DDT Map-Server receives the re-sent DDT Map-Request, with LISP-SEC information included, it decrypts the LISP-SEC information, performs normal LISP-SEC processing, and forwards the resulting Map-Request to the target ETR.

8.10. ETR sends Map-Reply to ITR

The ETR receives a Map-Request as documented in [LISP], performs any necessary processing of security information, as documented in [LISP-SEC], and sends a Map-Reply to the ITR that sent the original Map-Request.

9. Securing the database and message exchanges

This section specifies the DDT security architecture that provides data origin authentication, data integrity protection, and XEID prefix delegation. Global XEID prefix authorization is out of the scope of this document.

Each DDT node is configured with one or more public/private key pair(s) that are used to digitally sign referral records for XEID-prefix(es) that the DDT node is authoritative for. In other words, each public/private key pair is associated with the combination of a DDT node and the XEID-prefix that it is authoritative for. Every DDT node is also configured with the public keys of its children DDT nodes. By including public keys of target child DDT nodes in the Map-Referral records, and signing each record with the DDT node's private key, a DDT node can securely delegate sub-prefixes of its authoritative XEID-prefixes to its children DDT nodes.

Map-Resolvers are configured with one or more trusted public keys referred to as trust anchors. Trust anchors are used to authenticate the DDT security infrastructure. Map-Resolvers can discover a DDT node's public key either by having it configured as a trust anchor, or by obtaining it from the node's parent as part of a signed Map-Referral. When a public key is obtained from a node's parent, it is considered trusted if it is signed by a trust anchor, or if it is signed by a key that was previously trusted. Typically, in a Map-Resolver, the root DDT node public keys should be configured as trust anchors. Once a Map-Resolver authenticates a public key it locally caches the key along with the associated DDT node RLOC and XEID-prefix for future use.

9.1. XEID-prefix Delegation

In order to delegate XEID sub-prefixes to its children, a parent DDT node signs its Map-Referrals. Every signed Map-Referral also includes the public keys associated with each child DDT node. Such a signature indicates that the parent node is delegating the specified XEID -prefix to a given child DDT node. The signature is also authenticating the public keys associated with the children nodes, and authorizing them to be used by the children DDT nodes to provide origin authentication and integrity protection for further delegations and mapping information of the XEID-prefix allocated to the DDT node.

As a result, for a given XEID-prefix, a Map-Resolver can form an authentication chain from a configured trust anchor (typically the root DDT node) to the leaf nodes (Map-Servers). Map-Resolvers leverage this authentication chain to verify the Map-Referral

signatures while walking the DDT tree until they reach a Map-Server authoritative for the given XEID-prefix.

9.2. DDT node operation

Upon receiving a Map-Request, the DDT node responds with a Map-Referral as specified in Section 5. For every record present in the Map-Referral, the DDT node also includes the public keys associated with the record's XEID-prefix and the RLOCs of the children DDT nodes. Each record contained in the Map-Referral is signed using the DDT node's private key.

9.2.1. DDT public key revocation

The node that owns a public key can also revoke that public key. For instance if a parent node advertises a public key for one of its child DDT nodes, the child DDT node can at a later time revoke that key. Since DDT nodes do not keep track of the Map-Resolvers that query them, revocation is done in a pull model, where the Map-Resolver is informed of the revocation of a key only when it queries the node that owns that key. If the parent DDT is configured to advertise this key, the parent node must also be signaled to remove the key from the records it advertises for the child DDT node; this is necessary to avoid further distribution of the revoked key.

To securely revoke a key, the DDT node creates a new Record for the associated XEID-prefix and locator, including the revoked key with the R bit set. The DDT node must also include a signature in the Record that covers this record; this is computed using the private key corresponding to the key being revoked. Such a record is termed a "revocation record". By including this record in its Map-Referrals, the DDT node informs querying Map-Resolvers about the revoked key. A digital signature computed with a revoked key can only be used to authenticate the revocation, and should not be used to validate any data. To prevent a compromised key from revoking other valid keys, a given key can only be used to sign a revocation for that specific key; it cannot be used to revoke other keys. This prevents the use of a compromised key to revoke other valid keys as described in [RFC5011]. A revocation record must be advertised for a period of time equal to or greater than the TTL value of the Record that initially advertisied the key, starting from the time that the advertisement of the key was stopped by removal from the parent DDT node.

9.3. Map-Server operation

Similar to a DDT node, a Map-Server is configured with one (or more) public/private key pairs that it must use to sign Map-Referrals.

However unlike DDT nodes, Map-Servers do not delegate prefixes and as a result they do not need to include keys in the Map-Referrals they generate.

9.4. Map-Resolver operation

Upon receiving a Map-Referral, the Map-Resolver must first verify the signature(s) by using a trust anchor, or a previously authenticated public key, associated with the DDT node sending the Map-Referral. If multiple authenticated keys are associated with the DDT node sending this Map-Referral, the Key Tag field of the signature can be used to select the right public key for verifying the signature. If the key tag matches more than one key associated with that DDT node, the Map-Resolver must try verifying the signature with all matching keys. For every matching key that is found the Map-Resolver must also verify that the key is authoritative for the XEID-prefix in the Map-Referral record. If such a key is found, the Map-Resolver must use it to verify the associated signature in the record. If no matching key is found, or if none of the matching keys is authoritative for the XEID-prefix in the Map-Referral record, or if such a key is found but the signature is not valid the Map-Referral record is considered corrupted and must be discarded. This may be due to expired keys. The Map-Resolver can try other siblings of this node if there is an alternative node authoritative for the same prefix. If not, the Map-Resolver can query the DDT node's parent to retrieve a valid key. It is good practice to use a counter or timer to avoid repeating this process if the resolver cannot verify the signature after several trials.

Once the signature is verified, the Map-Resolver has verified the XEID-prefix delegation in the Map-Referral, and authenticated the public keys of the children DDT nodes. The Map-Resolver must add these keys to the authenticated keys associated with each child DDT node and XEID-prefix. These keys are considered valid for the duration specified in the record's TTL field.

10. Open Issues and Considerations

There are a number of issues with the organization of the mapping database that need further investigation. Among these are:

- Unlike in [LISP-ALT], DDT does not currently define a mechanism for propagating ETR-to-Map-Server registration state. This requires DDT Map-Servers to suppress returning negative Map-Reply messages for defined but unregistered XEID-prefixes to avoid loss of connectivity during partial ETR registration failures. Suppressing these messages may cause a delay for an ITR obtaining a mapping entry when such a failure is occurring.

- Defining an interface to implement interconnection and/or interoperability with other mapping databases, such as LISP+ALT.

- Additional key structures for use with LISP-DDT, such as to support additional EID formats as defined in [LCAF].

- Authentication of delegations between DDT nodes.

- Possibility of a new, more general format for the Map-Referral messages to facilitate the use of LISP-DDT with additional Key-ID/ IID/EID combinations. Currently-defined packet formats should be considered to be preliminary and provisional until this issue has received greater attention.

- Management of the DDT Map-Resolver referral cache, in particular, detecting and removing outdated entries.

The authors expect that experimentation on the LISP pilot network will help answer open questions surrounding these and other issues.

11. IANA Considerations

This document makes no request of the IANA.

12. Security Considerations

See Section 9 for a detailed description of protocol mechanisms intended to secure the database.

Open security issues include: xxx

13. References

13.1. Normative References

[LCAF] Farinacci, D. and J. Snijders, "LISP Canonical Address Format", draft-ietf-lisp-lcaf-06.txt (work in progress), October 2011.

[LISP] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-22.txt (work in progress), February 2012.

[LISP-MS] Fuller, V. and D. Farinacci, "LISP Map Server Interface", draft-ietf-lisp-ms-16.txt (work in progress), February 2012.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[RFC4634] Eastlake, D. and T. Hansen, "US Secure Hash Algorithms (SHA and HMAC-SHA)", RFC 4634, July 2006.

[RFC5011] StJohns, M., "Automated Updates of DNS Security (DNSSEC) Trust Anchors", RFC 5011, September 2007.

13.2. Informative References

[LISP-ALT] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "LISP Alternative Topology (LISP-ALT)", draft-ietf-lisp-alt-10.txt (work in progress), December 2011.

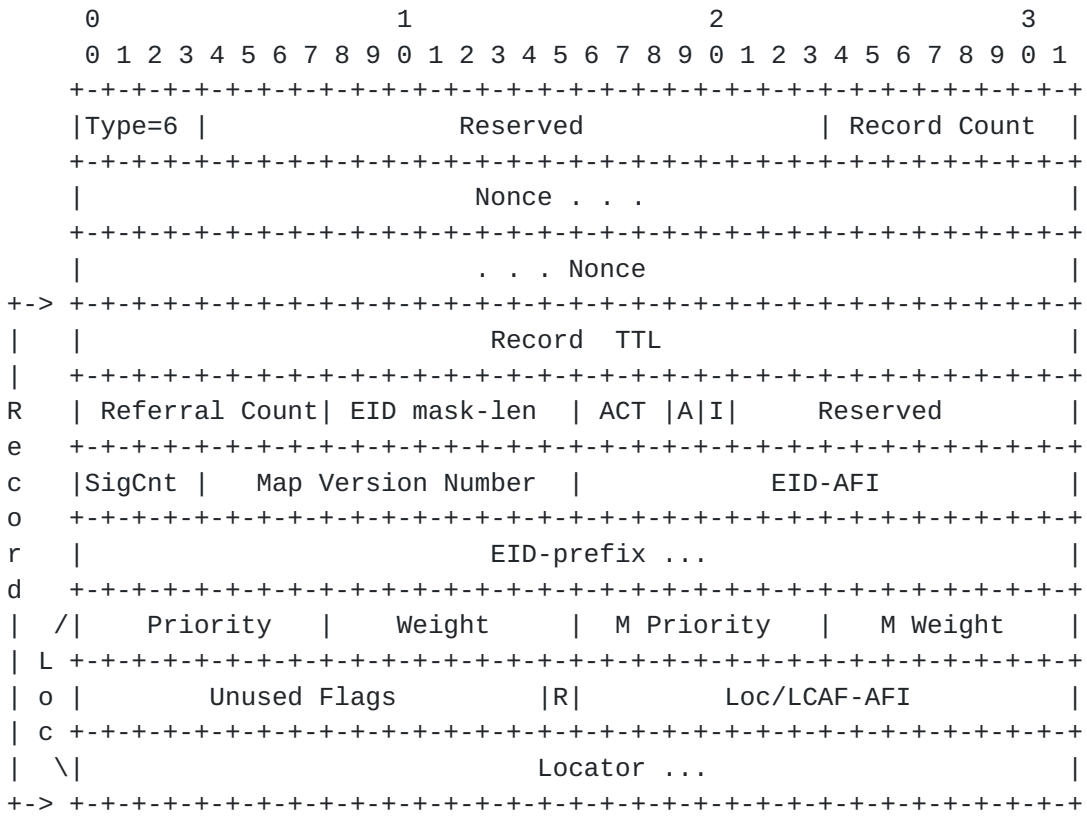
[LISP-SEC] Maino, F., Ermagan, V., Cabellos, A., Sanchez, D., and O. Bonaventure, "LISP-Security", draft-ietf-lisp-sec-01.txt (work in progress), January 2012.

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

Appendix A. Acknowledgments

The authors with to express their thanks to Damien Saucez, Lorand Jakab, and Olivier Bonaventure for work on LISP-TREE and LISP iterable mappings that inspired the hierarchical database structure and lookup iteration approach described in this document. Special thanks also go to Amit Jain, Isidor Kouvelas, Jesper Skriver, Andrew Partan, and Noel Chiappa, all of whom have participated in (and put up with) seemingly endless hours of discussion of LISP mapping database ideas and issues.

Appendix B. Map-Referral Message Format



ACT: The "action" field of the mapping record in a Map-Referral message encodes 6 action types. The values for the action types are:

Type 0, NODE-REFERRAL: Sent by a DDT node with a child delegation which is authoritative for the EID.

Type 1, MS-REFERRAL: Sent by a DDT node that has information about Map-Server(s) for the EID but it is not one of the Map-Servers listed, i.e. the DDT-Node sending the referral is not a Map-Server.

Type 2, MS-ACK: Sent by a DDT Map-Server that has one or more ETR registered for the EID.

Type 3, MS-NOT-REGISTERED: Sent by a DDT Map-Server that is configured for the EID-prefix but for which no ETRs are registered.

Type 4, DELEGATION-HOLE: Sent by an intermediate DDT node with authoritative configuration covering the requested EID but without any child delegation for the EID. Also sent by a DDT Map-Server with authoritative configuration covering the requested EID but for which no specific site ETR is configured.

Type 5, NOT-AUTHORITATIVE: Sent by a DDT node that does not have authoritative configuration for the requested EID. The EID-prefix returned MUST be the original requested EID and the TTL MUST be set to 0. However, if such a DDT node has a child delegation covering the requested EID, it may choose to return NODE-REFERRAL or MS-REFERRAL as appropriate. A DDT Map-Server with site information may choose to return of type MS-ACK or MS-NOT-REGISTERED as appropriate.

Incomplete: The "I" bit indicates that a DDT node's referral-set of locators is incomplete and the receiver of this message should not cache the referral. A DDT sets the "incomplete" flag, the TTL, and the Action Type field as follows:

Type	(Action field)	Incomplete	Referral-set	TTL values
0	NODE-REFERRAL	NO	YES	1440
1	MS-REFERRAL	NO	YES	1440
2	MS-ACK	*	*	1440
3	MS-NOT-REGISTERED	*	*	1
4	DELEGATION-HOLE	NO	NO	15
5	NOT-AUTHORITATIVE	YES	NO	0

*: The "Incomplete" flag setting on Map-Server originated referral of MS-REFERRAL and MS-NOT-REGISTERED types depend on whether the Map-Server has the full peer Map-Server configuration for the same prefix and has encoded the information in the mapping record. Incomplete bit is not set when the Map-Server has encoded the information, which means the referral-set includes all the RLOCs of all Map-Servers that serve the prefix. It is set when the Map-Server has not encoded the Map-Server set information.

SigCnt: Indicates the number of signatures (sig section) present in the Record. If SigCnt is larger than 0, the signature information

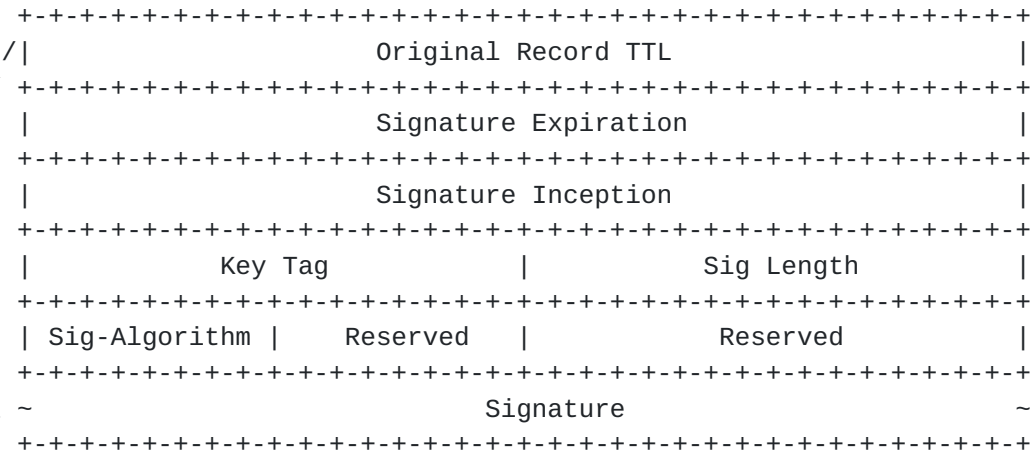
captured in a sig section as described in Appendix B.1 will be appended to the end of the record. The number of sig sections at the end of the Record must match the SigCnt.

Loc/LCAF-AFI: If this is a Loc AFI, keys are not included in the record. If this is a LCAF AFI, the contents of the LCAF depend on the Type field of the LCAF. Security material are stored in LCAF Type 11. DDT nodes and Map-Servers can use this LCAF Type to include public keys associated with their Child DDT nodes for a XEID-prefix referral record. LCAF types and formats are defined in [LCAF].

All the field descriptions are equivalent to those in the Map-Reply message, as defined in [LISP]. Note, though, that the set of RLOCs correspond to the DDT node to be queried as a result of the referral not the RLOCs for an actual EID-to-RLOC mapping.

B.1. SIG section

If SigCnt field in the Map-Referral is not 0, the signature information is included at the end of captured in a sig section as described below. SigCnt counts the number of sig sections that appear at the end of the Record.



Original Record TTL: The original Record TTL for this record that is covered by the signature. Record TTL is in minutes.

Key Tag: An identifier to specify which key is used for this signature if more than one valid key exists for the signing DDT node.

Sig Length: The length of the Signature field.

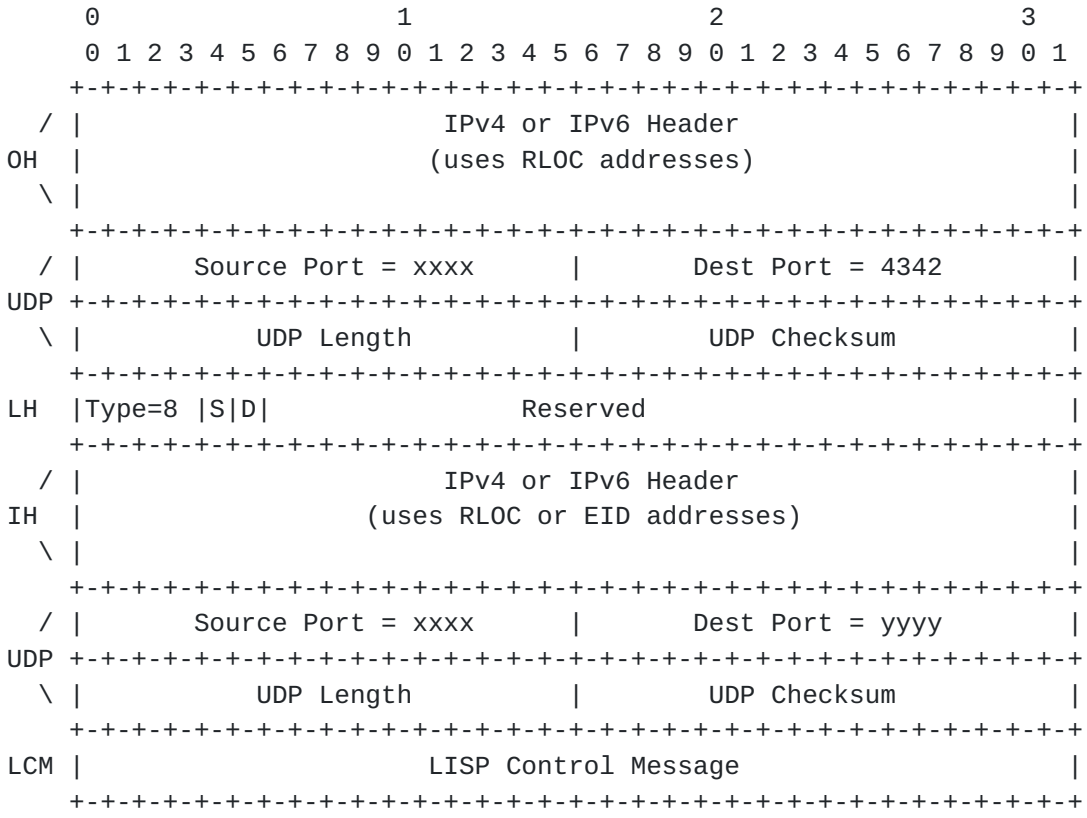
Sig-Algorithm: The identifier of the cryptographic algorithm used for the signature. Default value is RSA-SHA1.

Reserved: This field must be set to 0 on transmit and must be ignored on receipt.

Signature Expiration and Inception: Specify the validity period for the signature. Each specify a date and time in the form of a 32-bit unsigned number of seconds elapsed since 1 January 1970 00:00:00 UTC, ignoring leap seconds, in network byte order.

Signature: Contains the cryptographic signature that covers the entire record. The Record TTL and the sig fields are set to zero for the purpose of computing the Signature

Appendix C. Encapsulated Control Message Format



"D" is the "DDT-originated" flag and is set by a DDT client to indicate that the receiver can and should return Map-Referral messages as appropriate.

Authors' Addresses

Vince Fuller
cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: vaf@cisco.com

Darrel Lewis
cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: darlewis@cisco.com

Vina Eermagan
cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: vermagan@cisco.com