Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 24, 2014

F. Coras
A. Cabellos-Aparicio
J. Domingo-Pascual
Technical University of Catalonia
F. Maino
cisco Systems
D. Farinacci
lispers.net

October 21, 2013

LISP Replication Engineering
draft-coras-lisp-re-04

Abstract

This document describes a method to build and optimize inter-domain LISP router distribution trees for locator-based unicast and multicast replication of EID-sourced multicast packets.

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 http://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 April 24, 2014.

Table of Contents

1. Introduction

The Locator/Identifier Separation Protocol (LISP) [RFC6830] provides the mechanisms for the separation of Location and Identity semantics presently overloaded by IP addresses. The split results in the creation of two namespaces that unambigously identify edge-site network objects, Endpoint IDs (EIDs), and core routing objects, Routing LOCators (RLOCs). Apart from aiding the scalablity of the core routing infrastructure, the decoupling also enables the (re)implementation of new or existing inter-domain routing mechanisms.

One such mechanism is inter-domain IP source-specific multicast (SSM) [RFC4607]. In this sense, [RFC6831] defines the procedures carried out for delivering multicast packets from a source host in a LISP site to receivers residing in the same domain or in other LISP or non-LISP sites when an underlying multicast infrastructure exists. The signaling protocol it specifies for conveying (S-EID,G) state and building the distribution tree that connects the source ITR and the receiving ETRs is PIM [RFC4601]. An alternative method that uses Map-Requests for propagating (S-EID,G) state from ETRs to the ITR is established in [I-D.farinacci-lisp-mr-signaling].

Although desirable to use multicast routing in the core network when available, a mismatch between the multicast capabilities of receiver ETRs and source ITR might impede their interconnection. In such a case, unicast RLOC encapsulation is necessary to deliver multicast packets directly to the ETRs. This however leads to high ITR head-end replication for large sets of ETRs. Therefore, to reduce the replication load of the ITR and scale the service with the number of multicast receivers, the ITR may choose to offload replication to a set of RTRs.

The current document describes how multicast RTRs can be used to build an inter-domain distribution tree rooted at the ITR that can perform unicast and/or multicast encapsulated replication of multicast packets. This concept, of distributing the replication load from ITR to other RTRs downstream on the core overlay distribution tree, is known as Replication Engineering or LISP-RE. Since unicast replication in such overlays can be suboptimal when compared to the underlay network, methods to optimize packet delivery over the distribution tree are also presented.

This specification does not define the mechanisms used to build (S-EID,G) state in source and receiver domains, nor does it describe the messages used to propagate such state from receiver ETRs to source ITR. What it defines is how (S-EID,G) state is built in the ITR, RTRs and ETRs participating in the overlay distribution tree.

2. Definition of Terms

The terminology in this document is consistent with the definitions in [RFC6830] and [RFC6831] however, it is extended to account for LISP-RE concepts:

Delivery Group (DG): This is the outer destination address of a packet when LISP encapsulating a multicast packet with an EID source within a multicast packet.

Re-encapsulating Tunnel Router (RTR): An RTR is a router that implements the re-encapsulating tunnel function detailed in Section 8 of the main LISP specification [RFC6830]. Such router performs packet re-routing by chaining ETR and ITR functions, whereby they first remove the LISP header of ingressing packets and then prepend a new one prior to forwarding them.

Unicast Replication: Is the notion of replicating a multicast packet with an EID source address at an ITR or RTR by encapsulating it into a unicast packet. That is, the oif-list of a multicast map-cache entry can not only have interfaces present for link-layer replication and multicast encapsulation but also for site-facing interfaces and unicast encapsulation.

Overlay Distribution Tree: A degree-constrained spanning tree that represents the path followed by unicast and/or multicast encapsulated multicast packets from the root (ITR) to the leaves (ETRs) through intermediary nodes (RTRs). The ITR and RTRs unicast and/or multicast replicate packets to their tree children.

LISP Replication Node: A router (either the ITR or an RTR) participating and replicating packets downstream in the distribution tree.

Multicast Ingress Tunnel Router (ITR): An ITR as specificed in [RFC6830] that supports multicast and participates as the root in the distribution tree. In this document we use the term "ITR" to mean a multicast capable ITR.

Multicast Egress Tunnel Router (ETR): An ETR as specified in [RFC6830] that participates as a leaf in the distribution tree. ETR are the only members of the tree that do not unicast replicate. In this document we use the term "ETR" to mean a multicast capable ETR.

Multicast Re-encapsulating Tunnel Router (RTR): An RTR as specified in [I-D.farinacci-lisp-te] that participates as an intermediary node in the distribution tree. In this document we use the term "RTR" to mean a multicast capable RTR.

Replication Target: A multicast channel-id (S-EID,G) or a set of multicast channel-ids (S-EID-prefix,G).

Joining-OIF-list: Represents a collection of state per multicast routing table entry at an RTR or ETR that is created by received Map-Request/Join-Request.

Forwarding-OIF-list: Represents the outgoing RLOC list a multicast router stores per multicast routing table entry such that it knows to which RLOCs to replicate multicast packets. Although the Joining-OIF-list contains sufficient information to allow the forwarding of encapsulated multicast packets, using it is inefficient. Thereby, an RTR implementation may wish to build an efficient Forwarding-OIF-list. Ways of implementing a Forwarding-OIF-list are out of the scope of this document.

Upstream: Towards the root of the tree.

Downstream: Away from the root of the tree.

3. Overview

This document describes a method to diminish the ITR's replication load by using RTRs to build an inter-domain distribution tree. The tree is managed by the source domain's Map-Server. RTRs join the overlay due to either manual or automatic configuration and advertise to the Map-Server their availability to replicate traffic for a multicast channel (S-EID,G). Out of all the RTRs registering for the same multicast channel, the Map-Server builds one mapping and organizes the RLOCs in a multi-level hierarchy. The hierarchy is rooted at the ITR and computed based on the configured information RTRs register or by means of local policy and algorithms. ETRs always join the overlay as leaves and their attachment prompts the creation of a path, which traverses the RTR hierarchy, from the ITR. The path is built at receiver request by incrementally linking all distribution tree levels, starting at the joining ETR up to the source ITR.

The way the distribution tree is built has several benefits. First, it ensures that packets in the source domain do not reach the ITR if no ETR is joined. Second, it ensures that packets are forwarded from ITR to all ETRs without mapping database lookups since the state that

defines the distribution tree, i.e., the replication hierarchy, is created prior to forwarding/replicating the packets. Finally, the multicast source is allowed to roam since a first level RTR, when informed of the roam event, can do a new database lookup to find the new ITR to join to.

It is worth pointing out that because of the receiver-initiated approach multicast employs to build distribution trees, whereby receivers join upstream sources, LISP-RE operates backwards from LISP point of view. That is, ETRs are the ones to send Map-Requests to discover potential upstream parents and the ITR answers with Map-Replies to joining downstream clients.

4. Overlay Signaling

This section describes the signaling the ITR, RTRs and ETRs use in order to participate in the overlay and build a distribution tree. The signaling messages used are described in [I-D.farinacci-lisp-mr-signaling] and [RFC6831].

4.1. RTR Registration

RTR participation in the overlay is condition by the configuration of a replication target, a multicast channel (S-EID,G) or set of channels (S-EID-prefix,G), the RTR is to perform replication for. Once configured, manually or by automated mechanisms, an RTR Map-Registers its replication target with merge-semantics to the appropriate Map-Server. In the registration it also provides its list of RLOCs to be used by overlay peers and a set of corresponding weights and priorities. If present, information about the level of the hierarchy where the RTR should attach is also conveyed by means of an Replication List Entry canonical address [I-D.ietf-lisp-lcaf].

Due to the merge-semantics, the Map-Server aggregates all RTR originated Map-Register messages in a single, per replication target mapping. If no level information is provided or if so configured, the Map-Server should use local policy to compute a hierarchy and associate a level within it to each entry in the list (more details in Section 5.3). It should be noted that the entries that are pointed to in the resulting mapping are not RLOCs but Replication List entries.

4.2. ETR/RTR Subscription

When an ETR creates (S-EID,G) state from a site based multicast join, i.e., its oif-list goes non-empty, it must send an upstream Join request. If the ETR does not have multicast connectivity to its

upstream and unicast replication must be performed, the ETR requests that a path from ITR to itself, over the RTR hierarchy be constructed. The following procedure is followed to build the path:

1. ETR sends a Map-Request/Join-Request for (S-EID,G) multicast channel to the mapping database system which further ensures its delivery to the authoritative Map-Server.

2. The Map-Server looks up the mapping associated to (S-EID,G) and, out of the distribution tree hierarchy encoded within, it selects a set of leaf RTRs, i.e., members of the level furthest away from the ITR, with spare replication capacity. The set of potential parents is encoded in a new (S-EID, G) mapping the Map-Server conveys to the ETR in a Map-Reply.

3. From the list it receives, the ETR selects the best upstream RTR RLOC according to local policy, taking into account the associated priorities and weights and sends to the owning RTR a Map-Request/Join-Request for (S-EID,G). If the ETR itself has multiple RLOCs it wishes to use in the overlay, it may convey them all to the upstream RTR encoded in the Map-Reply field of the Map-Request/Join-Request together with associated priorities and weights.

4. The RTR stores the ETR's subscription information in the join-oif-list associated to (S-EID,G) and inserts the RLOC obtained after evaluating the priorities and weights in the oif-list for (S-EID,G). It then confirms the ETR's subscription with a Map-Reply.

5. If not already a member of (S-EID,G), the RTR initiates it's own attachment to the distribution tree by repeating the steps 1-4. An important difference at step 2, the Map-Server replies to a joining RTR with a list of RTRs in the adjacent upstream layer, as opposed to a list of leaf RTRs, like in the case of an ETR join. This procedure may recurse upstream up to when the ITR or an RTR already attached to the distribution tree is joined. On completion, there should exist a path from ITR to joining ETR.

6. If the ITR is already member of (S-EID,G) the process stops. Otherwise, the ITR sends a PIM join to the intra-domain multicast source ensuring the creation of a path from the multicast source to the receiver end-hosts.

If at any point, when creating a link between two adjacent layers, native multicast replication can be performed, instead of unicast replication, the router joining its upstream could set as source of the Map-Request/Join-Request a delivery group. However, group naming

must be coordinated between the participating parties in this case, if core network replication is to be exploited.

4.3. ETR/RTR Unsubscription

When an ETR's oif-list goes empty a Map-Request/Leave-Request is sent to the upstream RTR which will result in the removal of the ETR's associated entry from the RTR's oif-list. The procedure is repeated by the RTR, and it may recurse upstream, if its own oif-list also goes empty.

When an RTR with active dowstreams departs, it should first change the priority of the RLOCs it registers with the Map-Server to 255 and set its locators as unreachable in the RLOC-Probing replies it sends downstream. Finally, once all adjacent lower level members have sent Map-Request/Leave-Request messages the RTR can stop registering (S-EID,G) with the mapping database system and thus leave the overlay.

5. Overlay Management

5.1. RLOC Failure and Unreachability

RLOC failure is detected at control-plane level through RLOC-probing [RFC6830] by both upstream and downstream routers. When an RTR detects the failure of an downstream RLOC, it ceases replicating towards it. The affected RLOC is removed from the forwarding-oif-list and marked as unreachable in the join-oif-list. If a backup RLOC was provided by the downstream router in the Map-Request/Join-Request, it is instead inserted in the forwarding-oif-list and the failure results in no packet loss.

The routers downstream of a failed RTR RLOC, or who lost connectivity to said RLOCs, remove their Map-Request/Join-Request associated state and reperform the join procedure. Packet loss in this case must be solved by out-of-band mechanisms that are out of the scope of the current document.

5.2. Other Overlay Management Considerations

An overloaded RTR, i.e., one whose fan-out can not be increased, should change the priority of the RLOCs it registers with the mapping database system to 255. In such a situation, the Map-Server updates the associated mapping and informs all routers having requested it about the change through solicit Map Request (SMR) messages. Both new ETRs attaching to the distribution tree and those already connected but reperforming the join procedure must not use the RLOCs

with a priority of 255 as specified in [RFC6830]. However, routers having performed Join-Requests prior to the change should not break their existing connections to the affected RTR.

All routers part of an (S-EID,G) multicast channel should re-evaluate their attachment point to the distribution tree whenever the Map-Server updates the associated mapping. This ensures the overlay member routers attach to the best suited parent when new RTRs join or previously attached ones stop being overloaded. Change of a parent should be done following a "make before break" procedure. Specifically, the router changing attachment point first connects to the new parent and only afterwards sends the Leave-Request.

When a downstream RTR subscribes to a set of channel-ids (S-EID-prefix,G) using multiple RLOCs in a load-balancing configuration, the upstream RTR may choose to load-split channel-ids (S-EID,G) over the given set of RLOCs.

5.3. Automated Computation of RTR Level

Operators wishing to automate the RTR joining procedure may wish to use an algorithm for computing an optimized distribution tree. The algorithm could be implemented in the Map-Server and its output should be used to associate to all RTRs a level in the distribution tree. Due to the centralized management, on-line switching between algorithms may be possible in accordance to the required distribution tree performance. However, their use of such algorithms is dependent on the presence of overlay topological information. Ways of obtaining topological information will be discussed in future versions of this document.

5.3.1. Algorithm for Computing Optimized Distribution Trees

The current document does not recommend an algorithm for computing optimized distribution trees. However, it provides as an example a low computation cost heuristic, which, in the scenarios simulated in [LCAST-TR], can produce latencies between the ITR and the multicast receivers close to unicast ones. Its choice is to be influenced by operational requirements and the hardware constraints of the equipment in charge of running it. Future experiments might result in a recommendation.

In what follows, we use the term "distance" when referring to a relative length or amplitude of a metric, observed on a path connecting two points, but when the exact nature of the metric is of no interest.

Considering as goal the delivery of content for delay sensitive

applications, the function the algorithm minimizes is the maximum distance (e.g. latency or number of AS hops) from a multicast receiver to the ITR source. Notice that the reference is the multicast receiver host and not an ETR. Thus, what matters in deciding a member's position in the distribution tree is not solely its distance to the ITR but also the number of multicast receivers it serves. Then, a router close to the source but serving few receivers might find itself lower in the distribution tree than another with a slightly higher distance to the source but with a larger receiver set. The algorithm optimizes the quality of experience for multicast receivers and not for tunnel routers.

The problem described above, that searches for a minimum average distance, degree-bounded spanning tree (MADDBST), can be formally stated as:

Definition: Given an undirected complete graph G=(V,E), a designated vertex r belonging to V, for all vertices v in V, a degree bound d(v) <= dmax, dmax a positive integer, a vertex weight function c(v) with positive integer values, and an edge weight function w(e) with positive values, for all edges e in E. Let W(r,v,T) represent the cost of the path linking r and v in the spanning tree T. Find the spanning tree T of G, routed at r, satisfying that d(v) <= dmax and the distance to the source per multicast receiver is minimized.

The heuristic used to solve this problem works by incrementally growing a tree, starting at the root node r, until it becomes a spanning tree. For each node v, not yet a tree member, it selects a potential parent node u in the tree T, such that the distance per receiver to r, is minimized. At each step, the node with the smallest metric value is added to the tree and the parent selection is redone. The pseudocode of the heuristic is provided in Appendix A.

[SHI] and [BAN] have previously defined and solved similar optimization problems. Shi et al. [SHI] also prove that a particular instance of the problem, where all vertices have weight 1, is NP-complete for degree constraints 2 <= dmax <= |V|-1.

The algorithm can optimize an unicast overlay however, it should not be used to optimize multicast underlay delivery. As a result, if multicast is used as underlay between part of the overlay members, once one of the members of such Delivery Group is added to the distribution tree, the others should be marked as attached also. These nodes should receive multicast encapsulated multicast packets from the chosen node over the underlying multicast distribution tree.

Finally, since the RTRs do not replicate packets for multicast receiver hosts, prior to applying the MADDBST heuristic, a Minimum Spanning Tree (MST) algorithm should be used to compute the RTR distribution tree. In this case, the MADDBST heuristic should start attaching ETRs having as input the tree resulting from MST.

6. Security Considerations

Security concerns for LISP-RE the same as for [RFC6831] and [I-D.farinacci-lisp-mr-signaling].

7. IANA Considerations

This memo includes no request to IANA.

8. Acknowledgements

The authors would like to thank Noel Chiappa for his technical and editorial commentary.

9. References

9.1. Normative References

[I-D.farinacci-lisp-mr-signaling]
Farinacci, D. and M. Napierala, "LISP Control-Plane Multicast Signaling", draft-farinacci-lisp-mr-signaling-03 (work in progress), September 2013.

[I-D.farinacci-lisp-te]
Farinacci, D., Lahiri, P., and M. Kowal, "LISP Traffic Engineering Use-Cases", draft-farinacci-lisp-te-03 (work in progress), July 2013.

[I-D.ietf-lisp-lcaf]
Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", draft-ietf-lisp-lcaf-03 (work in progress), September 2013.

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

[RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for

IP", RFC 4607, August 2006.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

[RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.

9.2. Informative References

[BAN] Banerjee, S., Kommareddy, C., Kar, K., Bhattacharjee, B., and S. Khuller, "Construction of an efficient overlay multicast infrastructure for real-time applications", INFOCOM , 2002.

[LCAST-TR] Coras, F., Cabellos, A., Domingo, J., Maino, F., and D. Farinacci, "Lcast: Software-defined Inter-Domain Multicast", Technical Report http://personals.ac.upc.edu/fcoras/lcast-tr.pdf, 2012.

[SHI] Shi, S., Turner, J., and M. Waldvogel, "Dimensioning server access bandwidth and multicast routing in overlay networks", NOSSDAV , 2001.

Appendix A. MADDBST heuristic

```
INPUT: G = (V,E); r; dmax; w(u,v); c(v); u, v in V
OUTPUT: T

  FOREACH v in V DO
    delta(v) = w(r,v)/c(v);
    p(v) = r;
  END FOREACH

  T takes (U = {r}, D={});
  WHILE U != V DO
    LET u in U-V be the vertex with the smallest delta(u);
    U = U U {u}; L = L U {(p(u),u)};
    FOREACH v in V-U DO
      delta(v) = infinity;
      FOREACH u in U DO
        IF  d(u) < dmax and
            W{r,u,T} + w(u,v)/c(v) < delta(v) THEN
          delta(v) = W{r,u,T} + w(u,v)/c(v);
          p(v) = u;
        END IF
      END FOR
    END FOR
  END WHILE
```

Figure 1

Authors' Addresses

Florin Coras
Technical University of Catalonia
C/Jordi Girona, s/n
BARCELONA 08034
Spain

Email: fcoras@ac.upc.edu

Albert Cabellos-Aparicio
Technical University of Catalonia
C/Jordi Girona, s/n
BARCELONA 08034
Spain

Email: acabello@ac.upc.edu

Jordi Domingo-Pascual
Technical University of Catalonia
C/Jordi Girona, s/n
BARCELONA 08034
Spain

Email: jordi.domingo@ac.upc.edu

Fabio Maino
cisco Systems
Tasman Drive
San Jose, CA 95134
USA

Email: fmaino@cisco.com

Dino Farinacci
lispers.net

Email: farinacci@gmail.com