Network Working Group D. Meyer
Internet-Draft D. Lewis
Intended status: Informational Cisco
Expires: July 27, 2009 January 23, 2009

Architectural Implications of Locator/ID Separation
draft-meyer-loc-id-implications-01.txt

Status of this Memo

Copyright Notice

Abstract

Recent work on Locator/ID Separation has focused primarily on the control plane protocols concerned with finding Identifier-to-Locator

mappings. However, experience gained with a trial deployment of a system designed to implement Locator/ID Separation has revealed two general classes of problems that must be resolved after the mapping is found: The Locator Path Liveness Problem and the State Synchronization Problem. These problems have implications for the data plane as well as the control plane.

Table of Contents

1. Introduction

Locator/ID Separation (hereafter Loc/ID split) has been proposed as an architectural enhancement to the Internet architecture to facilitate, among other things, scaling of the global routing system [RFC1498][Chiappa99][Fuller06][RFC4984]. The basic idea is that the current number space (the IPv4/IPv6 address space) is overloaded with both location and identity semantics. One consequence of this overloading is that it is difficult to assign routing locators (RLOCs) in a way that is congruent with the underlying network topology; this makes aggregation difficult, if not impossible. This property is sometimes referred to as Rekhter's Law, and is frequently formulated as follows:

> "Addressing can follow topology or topology can follow addressing. Choose one."

Endpoint Identifiers (EIDs), on the other hand, are typically assigned without regard to the underlying network topology (for example, Host Identity Tags [RFC4423]). This makes it difficult for a single numbering space to efficiently serve both routing locator and endpoint identifier roles.

Locator/Identity Separation can be used to decouple the allocation of of EIDs from RLOCs, enabling the RLOC space to be aggregated aggressively (by aligning RLOC allocations with the underlying network topology). The positive effect of such aggregation would be to control the growth of global routing state. Note that aggregation in the EID space may also an issue, but as of this writing hasn't been explored extensively.

Recent work on Locator/ID Separation has focused almost exclusively on control plane protocols for finding Identifier-to-Locator mappings (for example, [I-D.fuller-lisp-alt][I-D.jen-apt] [I-D.lear-lisp-nerd]). However, experience gained with a trial deployment of a system designed to implement Locator/ID Separation has revealed two general classes of problems that must be resolved after the mapping is found: The Locator Path Liveness Problem and the State Synchronization Problem. These problems have implications for the data plane as well as the control plane.

This document focuses on the Locator Path Liveness and State Synchronization problems, and is organized as follows: Section 2

provides an overview of the problem space. Section 3 discusses the Locator Path Liveness problem, and Section 4 discusses the State Synchronization problem. Finally, Section 5 provides a few conclusions.

2. The Problem Space

Decoupling Location and Identity has profound implications both the control and data planes. In particular, decoupling location from identity leads to the two difficult problems: first, given a set of source locators and a set of destination locators, it must be possible to determine if a particular destination locator is reachable. We refer to this general problem as the Locator Path Liveness Problem. The Locator Path Liveness Problem is exhibited in host-based architectures such as SHIM6 [I-D.ietf-shim6-proto]) and HIP (Section 1.2 of [OPENHIP] describes an architecture in which "the failure detection daemon (reapd) is designed to be reused across HIP and shim6"), and network-based architectures such as RANGER [I-D.templin-ranger], eFIT [EFIT] and [LISP]. The "Hybrid Rewriting" class of architectures such as GSE [ODell97] exhibit a variant on the problem. Locator Liveness is discussed in detail in Section 3.

The second problem discussed in this document is that mapping state may need to be shared among network elements; this is as opposed to the determining if the locator itself is up or down. This is referred to as the Site-Based State Synchronization Problem, and is specific to network-based architectures. The Site-Based State Synchronization problem is discussed in Section 4.

3. The Locator Path Liveness Problem

The Locator Path Liveness Problem has been studied in various contexts [IANNONE08] [BARRE08] [OLIVA08] [OPENHIP] [I-D.ietf-shim6-failure-detection], and can be stated as follows:

> Given a set of source locators and a set of destination locators, can bi-directional connectivity be determined between the <source locator,destination locator> address pairs?

A simple example illustrates the problem. Consider the scenario depicted in Figure 1. Here a site S0 is multihomed to provider A and provider B. Further, suppose that S0 has a Provider Assigned (PA) locator from provider A (call it La) and a PA locator, Lb, from provider B. Suppose that provider A peers with provider B. In this case, S0 might "advertise" that its EID-prefixes can be reached through nodes La and Lb (either via DNS, explicit protocol message

such as a Map-Reply message [LISP], or other method) to its correspondent sites.

Now, suppose that a correspondent site S1 is connected to provider C, and that S0 has told S1 that it can reach S0 on either La or Lb.

Suppose further that S1 chooses La to reach S0, so that packets sourced from S1 destined for S0 traverse the path S1->C->B->A->S0. Note that if connectivity between provider B and provider A is disrupted, either for business or technical reasons, La will not be reachable from S1. In this case, S1 must detect that La is no longer reachable and use Lb to restore connectivity (in the event that S1 wants to restore connectivity; in today's Internet, S0 would continue to be unreachable).

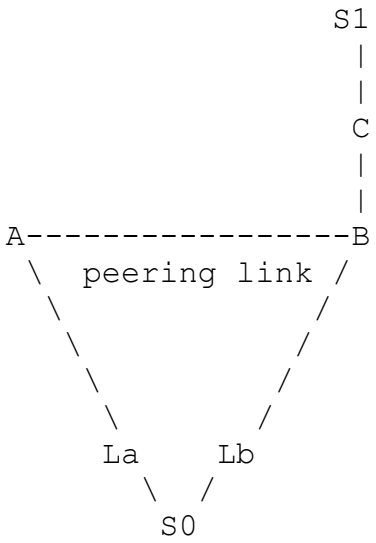

Figure 1: Reachability Failure

The Locator Path Liveness problem arises in subtly different ways, depending on the contents of the mapping database (i.e., EIDs, RLOCs, or some combination of these), who queries the database (host or network element), and how knowledge is distributed between hosts and routing elements. Note that in general, Locator Path Liveness must be tested in the data plane (although an implementation might take advantage of various "hints; see Section 3.3).

Host-Based Architectures: In host-based architectures (e.g., SHIM6 [I-D.ietf-shim6-proto]), the problem arises because queries to the database (DNS in this case) return "addresses" that can be thought of as a concatenation of the RLOC and EID. Because a host is anticipated to have multiple such "addresses" (at least in the SHIM6 case), it must choose a working <source,destination> pair from among its potential source addresses and its correspondent destination addresses. REAP [I-D.ietf-shim6-failure-detection] is

a probe-based reachability protocol that is designed to address this problem.

Hybrid Network-Based Rewriting Architectures: In hybrid network-based rewriting architectures (such as GSE [ODell97]), the problem arises because there is a knowledge asymmetry between the host and routing. Specifically, while the host is responsible for selecting the destination Routing Goop (RG) (i.e., the ingress point to the destination domain, essentially the destination RLOC), it is routing that selects the source RG. So while the IGP routing in a domain can be intelligent about egress points from the domain, it is the destination address, chosen by the host, that selects the ingress point in the destination domain.

This asymmetry gives rise to the following problem: Hosts will likely want information, at some granularity, about which <source,destination> pairs currently work. However, the host has no information about how many RGs are available to the site or if they are currently reachable. So the host cannot test the set of <source,destination> pairs for active paths. On the other hand, the routing system can't either, unless it snoops on TCP connections (which doesn't deal with asymmetric paths, UDP flows, or unidirectional flows). Section 4.2 of [Zhang06] discusses this issue from a slightly different point of view.

It is worth noting that unlike most "modern" descriptions of how GSE uses the DNS [Zhang06], the original GSE design [ODell97][ODell08] envisioned that the DNS would have a new resource record type, the RG record, to carry a site's RGs. Hosts would only have AAAA records. The idea was that for a given destination domain, a host in the source domain would compute the Cartesian Product {RGs}x{A4s}. Thus alternate path sensing would become a a matter of local policy, and not hard-wired by the destination domain (or whoever happens to be authoritative for the destination domain's names). Notice, however that even with the introduction of the RG resource record, the knowledge asymmetry remains.

Network-Based Map-and-Encap Archtectures: In the case of map-and-encap network-based architectures, the problem arises because the mapping element (e.g., Ingress Tunnel Router, or ITR) must choose among the RLOCs it has learned for a given EID-prefix. Thus an ITR can choose among the RLOCs associated with a given EID prefix, and a host may choose among multiple EIDs. However, a host cannot choose among the possible RLOCs; it simply has no access to that information (and even if it could, it would have no way to use

that information). Hence if the ITR chooses a RLOC that is not reachable, traffic to the destination site will be blackholed, and the host is left with no recourse.

3.1. The Multi-Exit Problem

The Multi-Exit Problem (MEP) arises when a site has two or more ITRs and there is a difference in destination RLOC reachability between the ITRs. For example, a site might have two ITRs, one which has connectivity to the destination RLOC, and one which doesn't. In this case, the host's packet will be carried by the site's internal routing (Interior Gateway Protocol, or IGP) to one of the exit points, as expected. However, since the IGP has no knowledge of locator liveness, and the host has limited ability to choose its exit (which may in any event be overridden by site routing), packets may be routed to a ITR that can not deliver them to the destination even though some ITR at the site can successfully deliver such packets.

As illustrated by the example above, the MEP arises because neither party (i.e., the host or the site's routing infrastructure) has both the knowledge or control necessary to detect the problem and route the packet accordingly. Note that while the MEP can arise without Locator/ID separation (for example, in the case in which site's border routers are taking default from their upstreams), the MEP can arise even when the site's routers have complete routing (e.g., a copy of the DFZ BGP table).

3.2. Complexity

The complexity of testing Locator Path Liveness in the data plane (i.e., probing) is roughly O(M*N), where there are M source addresses and N destination addresses. The following sections more closely analyze the complexity of host-based and network-based liveness probing. Note that the complexity described here is "worst-case". It is anticipated that implementations will develop heuristics such as those described in Section 3.3 to efficiently deal with Locator Path Liveness.

3.2.1. Complexity of Host-Based Probing

Host-based implementations must keep per-correspondent host liveness state. The complexity of probing in a host-based implementation can be thought of as follows:

```
Let C   = the number of correspondent hosts
Let D_i = the number of destination locators for host C_i
Let S   = the number of source locators
```

Then the complexity of host-based probing, P_host, is

O(P_host), where P_host = S*sum(D_i), i = 0...C-1

3.2.2. Complexity of Network-Based Probing

Network-based implementations must keep per-destination egress point liveness. The complexity of probing in a network-based implementation can be thought of as follows:

Let N = the number of EID-prefixes in a network element's cache
Let L_i = the number of locators for EID-prefix N_i
Let M = the number of source locators

Then the complexity of network-based probing, P_network, can be described as

O(P_network), where P_network = M*sum(L_i), i = 0...N-1

Note that a network-based probing scheme might have an advantage here because a single EID-prefix may cover many correspondent hosts. That is, sum(L_i), i = 0...N-1 < sum(D_i), i = 0...C-1

3.3. Possible Optimizations

The previous sections analyzed the complexity of explicitly probing to assess Locator Path Liveness. To mitigate this complexity, an implementation might rely on the various "hints" to assess Locator Path Liveness. The following sections, while not intended to be an exhaustive survey, outline some of the Locator Path Liveness hints an implementation may utilize.

Data Traffic: When data is received, an implementation might assume that the source of that traffic is reachable, and as such probing might not be needed. Of course, this is, at best, a unidirectional "hint" that an implementation might use to determine locator liveness. Only a complete round trip, wherein the distant site says something back to the local site which the local site originally sent to the distant site, can one then guarantee that the distant site can hear the local site.

A variation on this theme is to "piggyback" liveness testing on user data traffic, by adding a Solicit-User-Probe-Reply bit, that tells the far end to send back the next user data packet(s) with the outbound nonce, and a User-Probe-Reply bit set. Of course, this optimization depends on the existence of some traffic (even

if not for the same connection) going between pairs of border elements. That is, if a particular pair has only traffic in one direction, this method fails. In addition, it requires extra processing on user data packets, extra overhead in the packets (a field, some bits), and extra protocol complication. Of course,

such piggybacking only provides the view from remote domain, not whether the locator is actually reachable from the recipient of the "User-Probe-Bit".

Protocol Control Messages: If a protocol control message is received (for example, a Map-Reply), an implementation may conclude that the source of that is reachable. Again, in the best case, this is only a hint, because receipt of the control message proves only unidirectional connectivity.

Piggybacking Liveness Indications: A network-based architecture might piggyback indication of intra-domain locator liveness on other data and/or protocol messages. An example of this approach is LISP's use of loc-reach bits to indicate which Egress Tunnel Routers in a domain are up from the domain's perspective.

Existence of the Locator in underlying routing: A device which is responsible for locator liveness can utilize underlying routing to determine if the locator is at all available. If the network prefix (or a covering aggregate) for the destination locator is NOT found in underlying routing, then the path will not be available. This is at best a negative detection, it can show when a path is not available, but liveness of a particular locator. A given locator may still be unavailable and this not be shown in routing, due to data plane filtering, or the reachability being hidden by aggregation of the particular locator prefix.

Positive Feedback From Other Protocols: An implementation may be able to deduce some forms of reachability from other protocols. For example, TCP might indicate to the IP layer that it believes that there is bidirectional connectivity between a given address pair. This might be signaled to the source when it receives a SYN-ACK from the destination RLOC. As pointed out in [I-D.ietf-shim6-failure-detection], this is similar to how IPv6 Neighbor Unreachability Detection, which can be avoided when upper layers provide information about bidirectional connectivity [RFC4861].

If an implementation has access to higher layer protocols such as BGP, it might get a hint as to the reachability of a given locator. In the case of BGP, an implementation might conclude that the locator is reachable if there is a covering prefix in the BGP Routing Information Base (RIB). Again, this is a hint,

because the correspondent host may be down.

Timeouts: An implementation may be able to deduce some forms of Unreachability from timeouts of other protocols.For example, TCP might indicate that there is a lack of connectivity because it is not getting ACKs.Of course, this signal is overloaded: there may simply be congestion.

ICMP Messages: While ICMP is an available signalling protocol, due to its lack of security (in particular, ease of spoofing [I-D.ietf-tcpm-icmp-attacks]) and the fact that common policy is to block or rate limited ICMP, its utility has been somewhat marginalized (see Section 3.4). As such, ICMP may be used as a hint but beyond that, an implementation can not rely on ICMP as a signalling mechanism.

QQQ: Again, when do I know a locator is up? If I probe and the response is positive, does that mean its up (i.e., it can go down in the interim, so what is the time granularity, and what effect does that have on efficiency?

In general, depending on end-to-end liveness indications are applicable only to host-based solutions (e.g., [I-D.ietf-shim6-proto]). A network-based implementation may rely on higher layer protocols to indicate liveness (for example, an implementation may be able deduce a limited form of reachability from the existence of a BGP route covering the destination RLOC), but these too can only be used as hints. In the general case, however, an architecture that implements Loc/ID split (either host-based or network-based) will need to test Locator Path Liveness in the data plane

3.4. Security Issues

Mere inspection of insecure traffic may lead to false negative detection because of the insertion of malicious traffic. For instance, packets that masquerade as coming from a site may tamper with the loc-reach-bits, making the site's locators appear unreachable when in fact they are reachable [LISP].

ICMP Messages: ICMP messages are easily spoofable [I-D.ietf-tcpm-icmp-attacks], so they may be exploited to provide false negatives. However, they are also rate limited and often outright disabled, leaving a site sending data to a remote RLOC under the impression that the RLOC is reachable (a false positive

side effect of such filtering).

Existence of the Locator in the BGP RIB: This vulnerability is shared by non-Loc/ID split architectures (need reference to Pakistani-youtube example as a way compromised routing can break path liveness).

Aside from the ability to mislead a poorly implemented probing mechanism with data spoofing, probing creates a fundamentally unscalable relationship between site pairs (see Section 3.2). This leads to both implicit (unscalable) and explicit (vulnerable to probe floods) Denial of Service vulnerability in the systems receiving probe requests.

Finally, note that in the case of network-based Loc/ID separation architectures, the RLOCs of border elements represent reachability on behalf of entire site. As a result, failure to detect path liveness can disrupt connectivity to the entire site. On the other hand, in host-based Loc/ID separation architectures, only individual hosts are compromised.

4. Site-Based State Synchronization

The Site-Based State Synchronization problem is specific to network-based Loc/ID split architectures. There are two kinds of state synchronization that might need to be performed: mapping state synchronization and locator liveness synchronization.

The Site-Based State Synchronization problem can most easily be demonstrated by a simple example. Consider the following case: A site has two ITRs; one ITR is on the active path and the other ITR is on a backup path. In this case, all traffic egressing from the site traverses the ITR on the active path, and as a result that ITR is caching the mapping state for all of the active flows. The ITR on the backup path has no mapping state. Now, when the ITR on the active path fails, traffic is naturally shifted to the ITR on the backup path. If the now active ITR hasn't synchronized its state with the previously active ITR(s), then the newly active ITR has to reconstruct the mapping state for the flows that were traversing the failed ITR. In particular, the failure, which is local to the site, requires the now active ITR to go off-site to reconstruct the state.

4.1. Complexity

TBD

5. Conclusions

Architectures that implement Locator/ID Separation, either host or network based, need to evaluate carefully the complexity inherent in determining Locator Path Liveness. The complexity of mapping state synchronization is an additional concern for network-based architectures.

6. Acknowledgments

Shane Amante, Scott Brim, Noel Chiappa, John Day, Dino Farinacci, Vince Fuller, Mike O'Dell, Andrew Partan, and John Zwiebel provided insightful comments on early versions of this document. A special thanks goes to Mary Nickum for her attention to detail and effort in editing this document.

7. IANA Considersations

This document creates no new requirements on IANA namespaces [RFC2434].

8. References

8.1. Normative References

[Chiappa99]
Chiappa, N., "Endpoints and Endpoint Names: A Proposed Enhancement to the Internet Architecture", xxx 1999, <http://ana.lcs.mit.edu/~jnc//tech/endpoints.txt>.

[EFIT] Massey, D., "A Proposal for Scalable Internet Routing & Addressing", Feb 2007, <http://www.watersprings.org/pub/id/draft-wang-ietf-efit-00.txt>.

[Fuller06]
Fuller, V., "Scaling issues with ipv6 routing+ multihoming", Oct 2006, <http://www.iab.org/about/workshops/routingandaddressing/vaf-iab-raws.pdf>.

[I-D.fuller-lisp-alt]

Farinacci, D., "LISP Alternative Topology (LISP+ALT)",
draft-fuller-lisp-alt-02 (work in progress), April 2008.

[I-D.ietf-shim6-failure-detection]
Arkko, J. and I. Beijnum, "Failure Detection and Locator

Pair Exploration Protocol for IPv6 Multihoming", draft-ietf-shim6-failure-detection-13 (work in progress), June 2008.

[I-D.ietf-shim6-proto] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming Shim Protocol for IPv6", draft-ietf-shim6-proto-10 (work in progress), February 2008.

[I-D.ietf-tcpm-icmp-attacks] Gont, F., "ICMP attacks against TCP", draft-ietf-tcpm-icmp-attacks-03 (work in progress), March 2008.

[I-D.jen-apt] Jen, D., Meisel, M., Massey, D., Wang, L., Zhang, B., and L. Zhang, "APT: A Practical Transit Mapping Service", draft-jen-apt-01 (work in progress), November 2007.

[I-D.lear-lisp-nerd] Lear, E., "NERD: A Not-so-novel EID to RLOC Database", draft-lear-lisp-nerd-04 (work in progress), April 2008.

[I-D.templin-ranger] Templin, F., "Routing and Addressing in Next-Generation EnteRprises (RANGER)", draft-templin-ranger-00 (work in progress), October 2008.

[LISP] Farinacci, D., Fuller, V., Oran, D., and D. Meyer, "Locator/ID Separation Protocol (LISP)", draft-farinacci-lisp-11 (work in progress), Jan 2009.

[ODell08] Odell, M., "GSE - An Alternate Addressing Architecture for IPv6 (Private Communication)", Dec 2008.

[ODell97] Odell, M., "GSE - An Alternate Addressing Architecture for IPv6", Oct 2006, <http://www.watersprings.org/pub/id/draft-ietf-ipngwg-gseaddr-00.txt>.

[OPENHIP] Ahrenholz, J. and T. Henderson, "shim6 manual (html version)", 2007, <http://www.openhip.org/docs/shim6.html>.

[RFC1498] Saltzer, J., "On the Naming and Binding of Network

Destinations", RFC 1498, August 1993.

[RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[RFC4423] Moskowitz, R. and P. Nikander, "Host Identity Protocol (HIP) Architecture", RFC 4423, May 2006.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4984] Meyer, D., Zhang, L., and K. Fall, "Report from the IAB Workshop on Routing and Addressing", RFC 4984, September 2007.

[Zhang06] Zhang, L., "An Overview of Multihoming and Open Issues in GSE", Sept 2006, <http://www.cs.ucla.edu/~lixia/0609GSE_Overview.pdf>.

8.2. Informative References

[BARRE08] Barre, S. and O. Bonaventure, "Improved Path Exploration in shim6-based Multihoming", 2008.

[IANNONE08] Iannone, L., Saucez, D., and O. Bonaventure, "Implementing the Locator/ID Separation Protocol: Design and Experience", 2008.

[OLIVA08] de la Oliva, A., Bagnulo, M., Garcia-Martinez, A., and I. Soto, "Performance Analysis of the REAchability Protocol for IPv6 Multihoming", 2008.

Authors' Addresses

David Meyer
Cisco

Email: dmm@1-4-5.net

Darrel Lewis
Cisco

Email: darlewis@cisco.com