

# An Overview of Current IETF Activity on Routing and Addressing Models for the Internet

By David Meyer

## Abstract

The IAB's Routing and Addressing Workshop held in October 2006 in Amsterdam, rekindled interest in scalable routing and addressing architectures for the Internet. Among the many issues driving current interest are concerns about the scalability of the routing system and the imminent depletion of the IPv4 address space. One proposal, the Locator/ID Separation Protocol (LISP), is an experimental protocol being developed primarily by Cisco, and is intended to address (among other issues) the route scaling and address space depletion issues. This article is a summary of IETF activity in the area of Routing and Addressing Architectures for the Internet.

## I. Introduction

Since the IAB's Amsterdam workshop [1], several proposals have emerged that attempt to address concerns expressed both there and elsewhere [2].



In general, those proposals are based on the so-called *Locator/ID separation*, [3] which makes the assumption that separating the endpoint identification and routing locator functions of the IP address will lead to advantages for aggregatability (our only real tool to make the core routing system scale), mobility, and security.

Among the current data plane proposals circulating in the IETF are eFIT [4], LISP [5], and Six/One[6] (an interesting hybrid incorporating elements of shim6[7] and 8+8/GSE[8]). Note that these proposals seek a degree of incremental deployability, and in general they assume that the core routing system will not change. In addition, several of the proposals also require a system to map from "ID" to "Locator." Proposals in the mapping space (i.e., control plane proposals) include APT [9], LISP-ALT [21], LISP-CONS [10], and NERD [11].



Existing data plane proposals in this space leverage the one or more levels of indirection inherent in the Locator/ID separation idea to create one or more new namespaces. In most cases, two namespaces are utilized. One namespace—the Endpoint Identifiers (or EIDs)—is used to address hosts. The other space, known as Routing Locators (or RLOCs), is used for packet routing across a transit domain.

The goal of this indirection is to allow efficient aggregation in the RLOC space (which can be thought of as the current Default Free Zone, or DFZ) in order to provide persistent identity in the EID domain and, in some cases, to provide for secure and efficient mobility.

The remainder of this document is organized as follows: Section II discusses the various data plane proposals, and Section III overviews the control plane proposals. Section IV describes Cisco implementation status and testbeds. Finally, Section V offers some conclusions (and more questions).

## II. Data Plane Proposals

The current set of proposals fall into two broad categories: (1) map-and-encap and (2) address rewriting. The approaches differ depending on whether the translation occurs through address rewriting or tunneling and, in one case (Six/One), depending on where the indirection is implemented. The proposals are outlined as follows.

Cisco Confidential

Copyright © 2008 Cisco Systems, Inc. All rights reserved.

Page 1 of 6

### a. Map-n-encap

The general idea behind so-called map-and-encap (written *map-n-encap*) schemes, as originally described by Bob Hinden and Steve Deering, is that there are two address spaces: one used within a domain (the EID space) and one used to transit between domains (the RLOC space). The hope is that since the RLOC space is, in theory, decoupled from non-topologically assigned EID space, map-n-encap schemes will provide for efficient aggregation of the RLOC space—that is, the global routing state.

In the map-n-encap scheme, when a packet is generated, both its source and its destination “addresses” are taken from the site’s EID space. When a packet is addressed to a destination in another domain, it traverses the domain’s infrastructure to a border router (or other border element). The border router maps the destination of the EID to an RLOC, which corresponds to an entry point in the destination’s domain (hence the need for an EID-to-RLOC mapping system; mapping proposals are discussed later). This is the “map” phase of map-n-encap. The border router then encapsulates the packet and sets the destination address to the RLOC returned by the mapping infrastructure (if any; it may be statically configured as well). This is the “encap” phase of the map-n-encap model. Note that since map-n-encap works by appending a new header on an existing IP packet, it can work with both IPv4 and IPv6. While the destination EID is mapped to an RLOC in all of the proposals discussed here, the source EID in the packet may be treated differently within each proposal; specifically, it may or may not be mapped to an RLOC in the encapsulated packet. When the packet arrives at the destination border router, it is decapsulated and sent on to its destination. Note that this suggests that EIDs may need to be routable in some scope—most likely scoped to the local domain.

There are two map-n-encap proposals circulating in the IETF as of this writing<sup>1</sup>: Enable Future Internet Innovation through Transit Wire, or eFIT [12], and the Locator/ID Separation Protocol, or LISP [13]. The idea behind eFIT is that user networks and transit networks are separated in terms of both addressing and routing. User networks use EIDs, and transit networks use RLOCs. In eFIT, user and transit network routing domains are also separated. One of the interesting features of this proposal is that provider routing does not interact with routing in the user domains, which is different from Border Gateway Protocol (BGP), wherein user networks “peer” with provider networks using the same routing protocol and address space. In particular, there is no routing protocol operating across the links between the user networks and the transit core.

In contrast, LISP does not propose any classification of address spaces beyond the EID and RLOC spaces (more specifically, it has no concept of user or transit network spaces). Rather, in the LISP formulation, a site is assigned an EID prefix from which it addresses its hosts. When a host wants to send a packet to a remote domain, both the source and the destination in the packet contain an EID. At the domain boundary, routers do the same map-n-encap operation as described earlier.



Another major difference between LISP and eFIT is that LISP assumes there will be no changes to the core routing infrastructure. That is, LISP is transparent to the BGP infrastructure, whereas eFIT introduces boundaries between the user and the transit core networks that are not present in the current inter-domain (BGP) routing architecture. In particular, eFIT specifies that “There is no routing protocol operating across the links between the user networks and the transit core,” which represents a change from the current architecture.

---

<sup>1</sup> It is worth noting that there are several variations on the theme, including IPvLX (IP with virtual Link eXtension) [19] and Ivip (Internet Vastely Improved Plumbing) [20].

It is worth noting that map-n-encap schemes have the benefit of not requiring host changes or changes to the core routing infrastructure. However, there is some difference in opinion over whether the encapsulation overhead of map-n-encap schemes is problematic or not.

## b. Address Rewriting



The idea behind the address rewriting schemes—which were proposed originally by Dave Clark and later by Mike O’Dell [14]—is to take advantage of the 128-bit IPv6 address and use the top 64 bits as the routing locator (otherwise known as routing goop, or RG) and the lower 64 bits as the endpoint identifier<sup>2</sup>.

In this scheme, when a host emits a packet destined for another domain, the source address contains its identifier (frequently an IEEE MAC address) in the lower 64 bits and a special value (a unique “unspecified” value) in the RG. The destination address contains the fully specified destination address.

When a packet destined for a remote domain arrives at the local domain’s egress router, the source RG is filled in (forming a full 128-bit address) and the packet is routed to the remote domain. On ingress to the remote domain, the destination RG is rewritten with the unspecified value. This ensures that the host doesn’t know what its network prefix is and, as such, enables the renumbering that would be required to maintain the congruence between prefix assignment and physical network topology that is required for the kind of “aggressive envisioned” in the 8+8/GSE specification.

Six/One was the address rewriting approach presented at the RRG meeting. Six/One is interesting because it borrows ideas from both shim6 and 8+8/GSE. In particular, Six/One borrows the shim6 concept that multi-homed edge networks use provider-assigned addressing space from each of their providers and that hosts can use addresses from all of their providers interchangeably without breaking active transport sessions. Six/One borrows the 8+8/GSE idea of rewriting the high-order bits while packets are in flight. It also introduces the concept of edge networks. An edge network can independently route packets between two attached hosts, and predictably, edge networks connect to transit networks for global connectivity.

In Six/One, a host’s addresses differ only in their high-order bits (in much the same way as they do in 8+8/GSE). However, in a Six/One, an edge network (or other service provider) may change the address in a packet depending on the provider to which the packet is being routed. As a result, the destination address a host puts into a packet serves as a suggestion to the edge network about which provider the host’s packets should be routed to. The edge network may choose to either follow that suggestion or rewrite the high-order bits of the address in accordance with a provider of its own choice. Note that this is different than in shim6, where the host selects the transit network; in Six/One, edge networks retain the ability to select a particular provider via rewriting. Hosts adapt to address rewrites in that they modify subsequent packets accordingly before injecting them into the network. Unlike 8+8/GSE, Six/One also adds to packets certain information that enables the receiving hosts to reverse address rewrites.

What’s new about Six/One is that regardless of address changes, an edge network can also use the added information to identify a remote host. The main difference between Six/One and 8+8/GSE, then, is that hosts are aware of their full addresses (including the RG) and can suggest a network provider to their local domain (in the much same way that is enabled by the shim6 protocol). One of the many interesting aspects of the Six/One proposal is that it combines the host-based locator selection feature shim6 with a modified version of the

---

<sup>2</sup> While there is apparently quite a bit of interest in reviving 8+8/GSE, no new drafts have been published.

address-rewriting approach of 8+8/GSE. Finally, note that unlike the map-n-encap solutions described earlier, a Six/One host looks up the entire 128-bit address of the destination host in the DNS (which may return multiple AAAA records for the destination). Therefore, like shim6, no additional mapping system is needed.

Finally, it is important to note that address rewriting is really only practical for IPv6. This is important because middle boxes that need to filter can't do so if addresses continually change; in IPv6, we have enough bits to provide a stable EID (the lower 64 bits), so middle boxes can filter on EIDs.

### III. Control Plane Proposals

Since both map-n-encap and rewriting schemes rely on the addition of a level of indirection to the addressing architecture, it is necessary to map from the locally used address (EID) to the routing locator (RLOC). In the case of the map-n-encap schemes, it is a direct translation: an EID gets mapped to an RLOC. The situation is subtly different for the rewriting schemes: in general, such schemes must look up the entire destination address (which usually resides in the DNS) but it also must somehow find the source RG when rewriting the source address at a domain border. Six/One is a hybrid, since in that model the hosts know their entire address (including the RG), which can be looked up in the DNS, a property that is shared by shim6.

In the case of map-n-encap schemes, an EID-to-RLOC mapping service is required to make the service scale reasonably. There are three important parameters to consider in the creation of the architecture for a mapping service: the *rate* of updates to the database, the *state* required to be held by the mapping service, and the *latency* incurred during lookup. That is, a mapping system must minimize  $rate * state$  while still optimizing lookup latency. Because most estimates put the size of the mapping database at  $O(10^{10})$ , the implication is that the update rate must be small (note that this is a primary reason that current mapping proposals do not incorporate reachability information into the mapping database). In addition, the choice of push versus pull also has an effect on latency: if you push the entire database close to the edge, you improve lookup latency at the cost of increased state, and if you build a service that requires a mapping request in order to find an authoritative server for that mapping (in other words, pull), you reduce state in the core but you also increase latency. This suggests that a hybrid push/pull architecture might be the most effective. Regardless, architects need to take care not to import the dynamics (and hence the problems) of the routing system into the mapping database. If that were to happen, we wouldn't have solved the problem; we would have only moved it.

Four main proposals for mapping services have emerged: APT (A Practical Transit Mapping Service) [15], NERD [16], LISP-CONS (a Content Distribution Overlay Network Service for LISP) [17], LISP-ALT [21], and LISP-EMACS [22]. The proposals can be broadly classified as either push or pull (though both LISP-ALT and LISP-CONS might be considered hybrid protocols) based on how they distribute the database. LISP-EMACS is a multicast approach that uses multicast over a tunnel topology to distribute mappings.

Both APT and NERD are push protocols; that is, they *push* the mapping database to the edges for distribution. APT and NERD differ primarily (1) in how far toward the edge network the database is propagated (for example, APT has the concept of a default mapper so that some nodes can carry less than the complete database, whereas in NERD all nodes hold the complete database; in APT, the default mapper also winds up in the data path whenever it is used); (2) in database format<sup>3</sup>; and (3) in how the database is distributed and maintained (APT and LISP-ALT use BGP, and NERD uses HTTP).

---

<sup>3</sup> The APT database format isn't specified, LISP-CONS uses a (new) binary format, LISP-ALT uses BGP RIB format, and NERD uses XML.

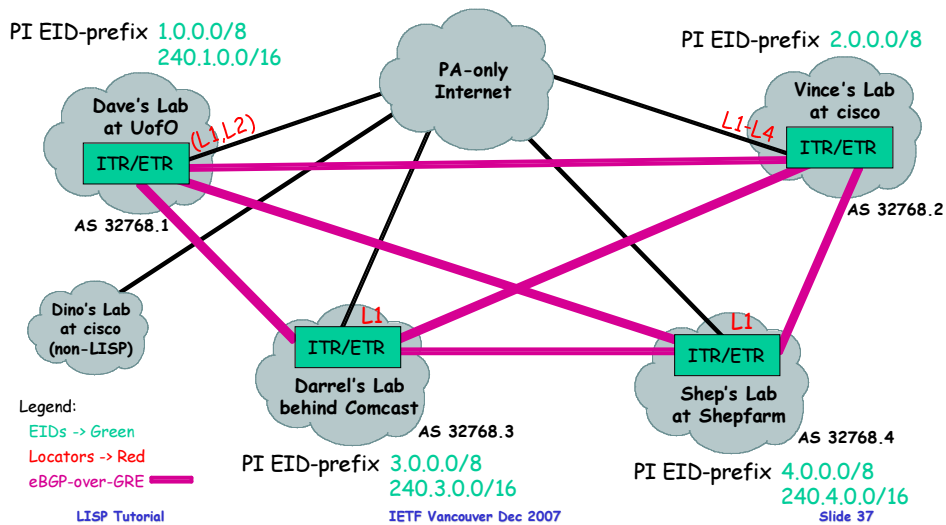
On the other hand, LISP-CONS and LISP-ALT are primarily pull protocols. That is, mappings must be *pulled* (via a query mechanism) from the authoritative servers. The actual EID-to-RLOC mappings reside in authoritative Content Access Resources (CARs), and mapping queries and replies traverse a hierarchical overlay from requester to the authoritative CAR (and back); in the case of LISP-CONS, a new protocol is used to route data probes and mapping requests and replies; LISP-ALT uses a separate BGP instance to accomplish the same functionality.

#### IV. LISP Implementation Status at Cisco



Cisco has aggressively pursued a prototype implementation on the DCOS "titanium" platform. To date, both the core LISP specification and LISP-ALT have been implemented. In addition, a LISP-ALT tested, shown below, is up and running.

### LISP-ALT Topology



If you would like to participate in the development of LISP, please subscribe to:  
[lispers@cisco.com](mailto:lispers@cisco.com)

#### V. Conclusions

Over the past 15 years, two major architectural approaches to the Locator/ID split have emerged: map-n-encap and address rewriting. While much progress has been made since the Amsterdam meeting, significant unresolved issues remain within all of the proposals, including the question of whether the Locator/ID separation solution is actually the best approach to a scalable Internet routing architecture. Other questions remain, such as whether map-n-encap schemes are superior to rewriting schemes such as 8+8/GSE. And what about host-based schemes, such as Six/One? How do these schemes interact with other protocols being developed in this space, such as shim6 or HIP[18]. Finally, since in most cases these schemes require another name resolution (ID to Locator lookup), there are questions about how best to build such a resolution system and whether such a system can be built in a scalable way that also is secure and minimizes latency.

Concerns about the scalability of the routing system, the effect of widespread deployment of IPv6 (especially given current RIR allocation policies), and the rapid depletion of the IPv4 "free pool" have fueled a growing interest in this area as well as in the broader topic of scalable routing and addressing architectures for the Internet. Much of this work is in the research phase (with LISP and LISP-ALT being the most fully specified and implemented), and more work needs to be done in the areas of interworking (transition), security and mobility (in the case of mobility, many people feel that mobility will be better handled below the network layer (e.g., cell phones) or above it (e.g., MIPv6). Finally, a deeper understanding of cost/benefit relationships, i.e., who bears the cost and who stands to benefit, is needed for all proposals.

## References

1. D. Meyer et al., "Report from the IAB Workshop on Routing and Addressing," RFC (Request for Proposal) 4984.
2. T. Narten et al., "Routing and Addressing Problem Statement," draft-narten-radir-problem-statement-01.txt.
3. N. Chiappa, "Endpoints and Endpoint Names: A Proposed Enhancement to the Internet Architecture," <http://ana.lcs.mit.edu/~jnc//tech/endpoints.txt>.
4. D. Massey, L. Wang, B. Zhang, and L. Zhang, "A Proposal for Scalable Internet Routing and Addressing," draft-wang-ietf-efit-01.txt.
5. D. Farinacci et al., "Locator/ID Separation Protocol (LISP)," draft-farinacci-lisp-05.txt.
6. C. Vogt, "Six/One: A Solution for Routing and Addressing in Ipv6", draft-vogt-rrg-six-one-01.txt.
7. E. Nordmark, "Shim6: Level 3 Multihoming Shim Protocol for IPv6," draft-ietf-shim6-proto-09.txt.
8. M. O'Dell, "GSE—an Alternate Addressing Architecture for IPv6," <http://www.watersprings.org/pub/id/draft-ietf-ipngwg-gseaddr-00.txt>.
9. D. Jen et al., "APT: A Practical Transit Mapping Service," draft-jen-apt-01.txt.
10. D. Meyer et al., "LISP-CONS: A Content Distribution Overlay Network Service for LISP," draft-meyer-lisp-cons-03.txt.
11. D. Massey, L. Wang, B. Zhang, and L. Zhang, "A Proposal for Scalable Internet Routing and Addressing," draft-wang-ietf-efit-00.txt. (Note: Not the same as item 4 above.)
12. Ibid.
13. Farinacci, op. cit.
14. O'Dell, op. cit.
15. Jen, op. cit.
16. E. Lear, "NERD: A Not-So-Novel Eid-to-Rloc Database," draft-lear-lisp-nerd-02.txt.
17. Meyer, op. cit.
18. R. Moskowitz et al., "Host Identity Protocol," draft-ietf-hip-base-10.txt.
19. Whittle, Robin, "IP Vastly Improved Plumbing (IVIP)", <http://www.fistpr.com.au/ip/ivip>
20. Templin, F. (Ed), "The IPvLX Architecture", draft-templin-ipvlx-08.txt.
21. Fuller, V., et. al., "LISP Alternate Topology", draft-fuller-lisp-alt-01.txt.
22. Brim, S., et. al., "EID Mappings Multicast Across Cooperating Systems for LISP", draft-currans-lisp-emacs-00.txt.