CoreSim: A Simulator For Evaluating Locator/ID Separation Protocol Mapping Systems

Florin Coras
Technical University of Cluj-Napoca
florin.coras@net.utcluj.ro

Loránd Jakab
Universitat Politècnica de Catalunya
ljakab@ac.upc.edu

Albert Cabellos-Aparicio
Universitat Politècnica de Catalunya
acabello@ac.upc.edu

Jordi Domingo-Pascual
Universitat Politècnica de Catalunya
jordid@ac.upc.edu

Virgil Dobrota
Technical University of Cluj-Napoca
Virgil.Dobrota@com.utcluj.ro

ABSTRACT

The alarming growth of the default free zone (DFZ) routing table prompted a response from the routing research community, which replied with several options, many of them centering around the idea of core-edge separation. The Locator/ID Separation Protocol (LISP) is one of the most promising proposals, which gained momentum recently. Some implementations exist, with a small experimental testbed, but there are no tools to evaluate how these proposals would fare Internet-scale deployments. We propose CoreSim, an extensible LISP simulator, which can be used to evaluate the performance of this protocol considering today's Internet topology.

1. INTRODUCTION

The 2006 Internet Architecture Board Routing and Addressing Workshop report [1] listed the alarming rate of growth of the Default Free Zone (DFZ) routing table the most important problem facing the Internet today. The scalability issues of the current Internet Routing architecture were previously recognized, and proposals already existed for future Internet architectures. Many of them are centered around the idea of separating the network node's identifier from its topological location. This report however sparked research for a solution, that would enable incremental deployment of the new proposed protocol, changing as little as possible current hardware, software and addressing schemes. The Locator/ID Separation Protocol (LISP) [2] is a promising proposal, pushed by Cisco and academia, that tries to meet the above mentioned goals.

In LISP, each end-host has two addresses: an Endpoint Identifier (EID) that is related with the identity of the node, and a Routing Locator (RLOC) that it is related with the topological position of the network where it is attached to. The EID is only routable inside the domain while the RLOC outside the domain. In order to provide bindings between EIDs and RLOC, LISP employs a special distributed database called Mapping System (MS).

Work on LISP and MS proposals is currently underway in the LISP WG of the IETF. The main protocol and LISP+ALT, one of the mapping systems, already have an experimental implementation for Cisco IOS, deployed in a testbed of about 20 nodes (lisp4.net). An open source implementation called OpenLISP [3] also exists for the FreeBSD kernel.

These implementations help validate the good functioning of the proposals, especially at micro level. Nevertheless, to evaluate them at macro level, to see how they scale to the size of the current Internet and if they are feasable, a simulator can be a useful complementary tool for the LISP research community. To the best of the author's knowledge there is no such previous work.

We have created CoreSim, an Internet-scale LISP deployment simulator[1]. It is able to replay a packet trace and simulate the behaviour of a LISP Ingress Tunnel Router (ITR) and the associated Mapping Resolver, on top of a topology based on measurements performed by the iPlane infrastructure (described in the next section). It reports mapping lookup lantency, the load imposed on each node of the MS and cache performance statistics. We have LISP+ALT and LISP-DHT implemented so far, but other MS can be added in the future. The simulator will soon be made public under the GPL license.

2. THE SIMULATOR

CoreSim has two main building blocks (see figure 1): one simulates a LISP ITR with its associated operations (sending Map-Requests, buffering packets, caching Map-Replies, and forwarding packets), the other the mapping system (path taken and latency of the lookup).

[1] Available from http://www.cba.upc.edu/lisp

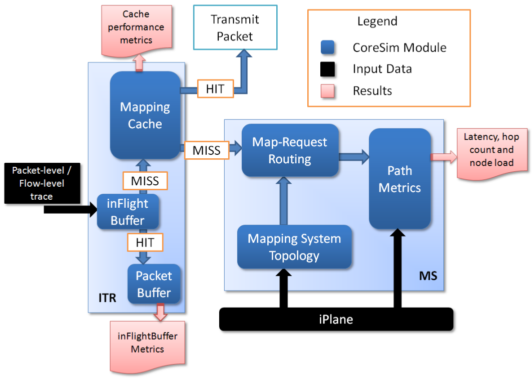

Figure 1: CoreSim architecture

Topology. The foundation of the simulator is the iPlane platform. It combines BGP looking glass information with a very large number of daily traceroutes from over 400 vantage points in order to provide Internet topology information. At its heart, iPlane is an Internet latency lookup service, returns an estimated latency between arbitrary IP addresses. For the datasets we used for lookups we obtained an approximately 65% success rate. For the remaining 35% of cases we developed an estimator based on the geographical distance between nodes, using successful lookups as training data, and linear regression as the algorithm.

The topology nodes used in the simulator are points of presence (PoPs) as defined by iPlane measurements. Each PoP has several IP addresses belonging to it and links to other PoPs are also defined. This information is used to select LISP routers for our topology, as follows. First, we determine the AS of the PoPs, and count the number of links that it has to PoPs in other ASes (inter-domain links). We consider as the AS border router and LISP tunnel router (xTR) the PoP with the highest number of inter-domain connections.

ITR. The packet trace is fed to the ITR module, which upon inspecting a packet's destination address looks up the local mapping cache and "forwards" the packet in the case of a cache hit. A miss triggers a lookup using the MS module, and the packet is buffered until the Map-Reply is returned.

Mapping System. Each of the mapping systems is built on top of the iPlane topology based on its specific architecture. The simulator routes Map-Requests from the node associated to the ITR to the node responsible for the mapping. The path of the query is recorded and the respective latencies added together, to compute path length (number of hops and time) and save statistics of number of queries routed by individual nodes.

3. RESULTS

We validated our implementation of the Chord protocol for LISP-DHT using the LISP-DHT implementation developed at Universite catholique de Louvain, which is based on OpenChord 1.0.5. For a preliminary evaluation of our LISP+ALT and LISP-DHT implementations we used a 35 second packet trace from our university's Internet link, consisting of 2 million packets. Due to space constraints we only present the graph for the packet buffer evolution, see figure 2.

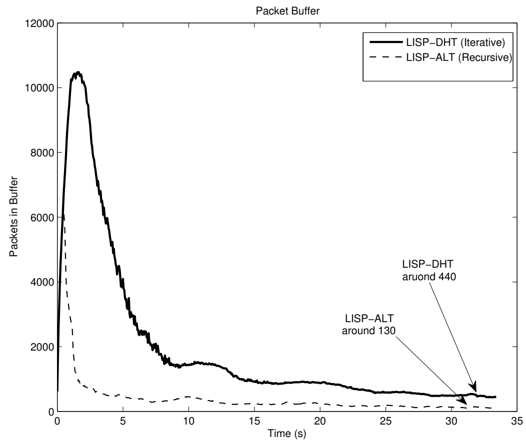

Figure 2: Packet buffer evolution

There is a short initial transient period of about 15 seconds, while the ITR cache is being "warmed up". After that the buffer seems to stabilize around 130 packets consisting of 15 KB of data for LISP+ALT and 440 and 40KB for LISP-DHT respectively. We are working on more complete analysis of buffer behaviour. Regarding the lookup latency, our preliminary analysis shows that it is below 1 s for 90% of the cases in LISP+ALT, but LISP-DHT almost always is more, with a median just below 3 s. These latencies are added up by 5 hops in LISP+ALT and 10 in LISP-DHT (median).

4. SUMMARY AND FUTURE WORK

We described CoreSim, the first Internet-scale LISP simulator, which works on unidirectional packet traces and simulates a LISP ingress tunnel router. In the future other mapping system proposals could be considered. Implementing several different ITRs is also being discussed.

5. REFERENCES

[1] D. Meyer et al. "Report from the IAB Workshop on Routing and Addressing." RFC 4984, Sept. 2007.
[2] D. Farinacci et al. "Locator/ID Separation Protocol (LISP)." IETF draft. Mar. 2009
[3] OpenLISP: http://inl.info.ucl.ac.be/softwares/openlisp
[4] iPlane: http://iplane.cs.washington.edu/