Can TCP and Locator/ID Separation get along?

Juhoon Kim
Deutsche Telekom Laboratories
Technische Universität Berlin
jkim@net.t-labs.tu-berlin.de

Luigi Iannone
Deutsche Telekom Laboratories
Technische Universität Berlin
luigi@net.t-labs.tu-berlin.de

Anja Feldmann
Deutsche Telekom Laboratories
Technische Universität Berlin
anja@net.t-labs.tu-berlin.de

ABSTRACT

Aiming at solving the scalability problem that the current Internet is facing, the separation of the IP namespace into two different namespaces (the Locator and the Identifier namespaces) is one of the most promising paradigms. However, the impact of this new paradigm on the Internet traffic is yet to be assessed. In this extended abstract, we present a preliminary analysis of TCP performance in the context of Locator/ID separation.

1. INTRODUCTION AND MOTIVATION

Due to multi-homing, traffic engineering, and provider independent address assignment, the current Internet is suffering from several scalability problems as discussed in [1]. The ever-growing consumption of IP addresses comes along with the increase of the routing table size in the core Internet, caused by the IP namespace being more finely fragmented. Further, it is commonly believed that this phenomenon will be exacerbating when IPv6 will be deployed. Since this scalability problem increases costs for network operators, numerous studies have been made to reduce the routing table size of the core Internet. It is commonly accepted in the research community that separating the current IP namespace into the router locating namespace (Routing LOCator, a.k.a., RLOC) and the end-host identifier namespace (Endpoint IDentifier, a.k.a., EID) [2] will alleviate the issue.

Although several sibling protocols have been introduced within this architecture, the Locator/ID Separation Protocol (LISP [3]) is the most actively developed technology at the moment. In LISP, packets are tunneled through the core Internet from the edge border router of the source site to that of the destination site. To this end, the Ingress Tunnel Router (ITR) must know which Egress Tunnel Routers (ETRs) are responsible for the destination end-host. Such information can be learned from the mapping system, which provides the binding information between RLOCs and EIDs upon router's request. LISP border routers store the mapping information retrieved from the mapping system in a local cache to use them for subsequent packets. In this way, only the initial packet to a destination site needs a lookup in the mapping system.[1]

However, it is unavoidable that in case of cache-miss the initial packet is delayed or retransmitted from the source end-host, causing performance degradation of Internet services. In a previous study [4], we have shown the scalability properties of the LISP Cache, including the impact of the cache timeout on the cache-miss rate. To explore that further, we are currently focusing on the impact of the initial cache-miss on Internet traffic. We focus our analysis on TCP since the large majority of Internet traffic is delivered over TCP. For this purpose we deployed a testbed in order to carry out traffic observation and evaluate the impact of LISP on the performance.

[1]In practice, a cache entry should be removed when it is not used for a certain period of time. Thus, in the long run, multiple cache-misses may be caused by one destination site.

Table 1: TCP parameters used in the testbed

Parameter	Value
Initial congestion window	4 MSS
Delayed ACK time	100 ms
Retransmission TimeOut	3000 ms

2. EXPERIMENTAL MEASUREMENTS

In order to perform our measurement in a real LISP environment, we create a testbed as illustrated in Figure 1. FreeBSD 8.0 is used as an operating system of all machines in the testbed. Vanilla FreeBSD is installed in both source and destination end-hosts, while OpenLISP [5] enhanced FreeBSD is installed on ITR and ETR. Furthermore, for ITR and ETR, we wrote software able to emulate the mapping system with tunable query/reply latency, in order to be able to explore the impact of the mapping latency. We generate TCP traffic using iPerf and produce the propagation delay using Dummynet. In our experiments, we use default value of FreeBSD 8.0 for TCP parameters (see Table 1).

The experiments are conducted in three different scenarios: the normal Internet technology and two LISP strategies. The first LISP strategy is vanilla LISP, *i.e.*, EID-to-RLOC mapping lookup and caching being performed only in the source site. The second strategy is symmetric LISP, *i.e.*, the mapping lookup and caching being performed both in the source and the destination sites due to security concerns (see also [4]). In each scenario, we carry out the experiments

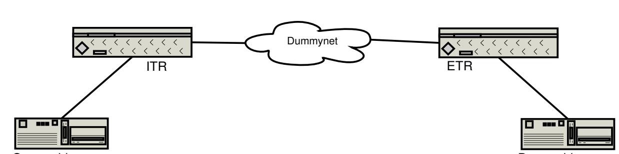

Figure 1: LISP testbed

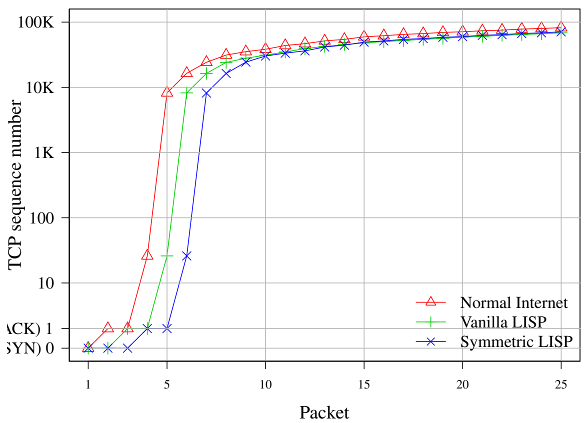

Figure 2: Sequence numbers

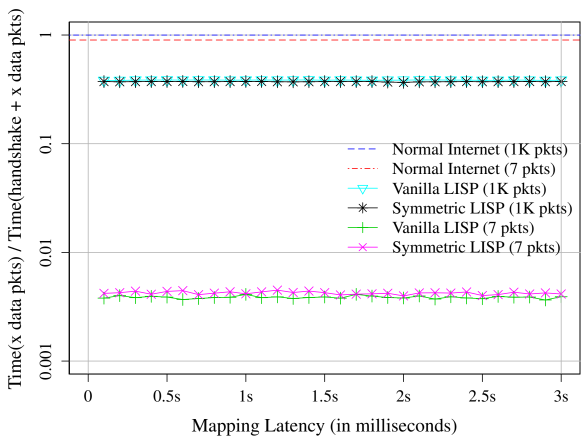

Figure 3: Performance comparison

changing the mapping latency from 100 ms to 3000 ms, with 100 ms step. For each latency value, we repeat the experiment 100 times, while collecting traffic from both source and destination end-hosts.

3. TCP OVER LISP

In order to show what happens when the initial packet of a TCP flow causes a cache-miss, we illustrate the sequence number of first 25 packets of source-to-destination flows for both vanilla LISP and symmetric LISP as well as for the normal Internet case (*cf.*, Fig. 2). Each point of the figure represents an average value of 100 experiments. We can observe one SYN packet retransmission for vanilla LISP and two packet retransmissions for symmetric LISP. More precisely, the first packet for vanilla LISP is dropped in the border router of the source site, as the ITR does not have the EID-to-RLOC mapping in its local cache. The same happens for symmetric LISP, however, in this case the second packet is again dropped (this time in the destination site), since the ETR also needs a mapping. Since the current LISP implementations drop packets that cause cache-miss, the latency delay due to cache-misses can be estimated by multiplying the number of cache-misses by the TCP Retransmission TimeOut (RTO).

To evaluate the performance of the whole TCP flows we calculate the ratio between the pure data transfer time and the total connection time (hence including the handshake). To see the performance difference between short flows and long flows, we performed measurements for flows transferring only 7 packets and for flows transferring 1000 packets (as defined in [6]). Results are presented in Fig. 3, where we can see that the ratio is close to 1 for both short and long flows in the normal case without losses, while in the case of LISP, it shows a significantly lower ratio. Further, the difference between long flows and short flows is much bigger in the context of LISP than in the normal Internet. Given the fact that short flows contribute more than 90% of all global TCP flows [7], this characteristic is not negligible.

4. SUMMARY AND ONGOING WORK

The analysis we presented aims at shedding light on the impact of solutions like LISP have on the Internet traffic. Our observation confirms that the cache-miss behavior of LISP causes significant degradation of TCP performance. Our current research is more directed at determining how to optimize the TCP parameters for Locator/ID split technologies. Although the increase of the TCP traffic latency due to the initial cache-miss in the LISP-enabled Internet mainly depends on the RTO, finding an optimal value is a non-trivial task. Indeed, the insufficient RTO, interacting with other TCP parameters such as delayed ACK time and initial congestion window size, may decrease the performance of TCP traffic [7]. In addition, the optimization must not hinder the communication between a TCP speaker in LISP-enabled site and a normal TCP speaker. For our ongoing research, we take all these considerations into account to determine the optimal TCP parameters.

5. REFERENCES

[1] D. Meyer, L. Zhang, and K. Fall, "Report from the IAB Workshop on Routing and Addressing," September 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4984.txt

[2] T. Li, "Recommendation for a Routing Architecture," February 2011. [Online]. Available: http://www.ietf.org/rfc/rfc6115.txt

[3] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "Locator/ID separation protocol (LISP)," Internet Engineering Task Force, Internet Draft (Work in Progress) *draft-ietf-lisp-15.txt*, July 2011.

[4] J. Kim, L. Iannone, and A. Feldmann, "A deep dive into the lisp cache and what isps should know about it," in *IFIP International Conference on Networking*, May 2011, pp. 267–278.

[5] The OpenLISP Project. [Online]. Available: http://www.openlisp.org

[6] A. Hafsaoui, D. Collange, and G. Urvoy-Keller, "Revisiting the performance of short tcp transfers," in *IFIP International Conference on Networking*, May 2009, pp. 260–273.

[7] N. B. Azzouna and F. Guillemin, "Analysis of adsl traffic on an ip backbone link," in *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE*, December 2003, pp. 3742–3746.