# LISP Mapping System as DoS Amplification Vector

Mattias Gabriel
*Montefiore Institute*
*Université de Liège*
Liège, Belgium
mattias.gabriel@student.uliege.be

Luigi Iannone
*Datacom Department*
*Huawei Technologies Co. Ltd*
Paris, France
luigi.iannone@huawei.com

Benoit Donnet
*Montefiore Institute*
*Université de Liège*
Liège, Belgium
benoit.donnet@uliege.be

***Abstract*—There is a growing interest in solutions relying on the identifier/locator separation paradigm. It introduces several benefits in terms of scalability and flexibility. It relies on two addressing spaces, namely the *identifiers*, for endpoint identification, and the *locators*, for packet forwarding. An additional control plane is necessary to map one space to the other.**

**In this paper, we explore how control messages can be an amplification vector for DoS attacks. We evaluate the possible amplification factor based on a real deployment, showing that the amplification factor exists. We also build a GNS-3 testbed to demonstrate further and analyze the attack.**

***Index Terms*—LISP, Mapping System, Amplification, IP Spoofing, DoS**

## I. INTRODUCTION

In recent years, both academia and industry have explored ways to enhance the Internet Architecture [1], with ongoing efforts imagining the Internet ten years from now [2], [3]. One of the cornerstones retained to achieve such a goal is to leverage on the locator/identifier separation paradigm because of the benefits and flexibility such an approach introduces in the hourglass architecture of the Internet [4], [5].

The locator/identifier separation paradigm relies on the existence of two different address types: the *identifiers* and the *locators*. An identifier is used to identify a connection endpoint and is only locally routable. In contrast, a locator refers to a node attachment point in the Internet topology and is globally routable. In order to perform data plane operations, it is necessary to be able to associate identifiers with locators. Thus, an indirection mechanism is necessary, binding identifiers to locators or, stated differently, gluing the identifier addressing space to the locator addressing space. Such an indirection mechanism is generally named a Mapping Distribution System (MDS), which is queried in an on-demand fashion, similarly to the DNS system.

This paper addresses an important issue in the locator/identifier separation paradigm, notably the (ab)use of the MDS control messages as an amplification vector for a denial-of-service attack. The principle of such a kind of attack is quite simple and somewhat equivalent to the DNS amplification attack [6]: the attacker forges a message (the source IP address being spoofed with the one of the targeted victim), for querying the MDS to get locators associated to identifiers. Politely, the MDS replies to the victim, with a set of locators, this message being possibly larger than the one sent by the attacker. With the appropriate level of bandwidth, the attacker has the potential to drown the victim with undesired messages.

This paper makes the following contribution. We use LISP (Locator/Identifier Separation Protocol [7]) as a tool to obtain a realistic evaluation. We build a GNS-3 simulated testbed for demonstrating the attack. To the best of our knowledge, we are the first to showcase locator/identifier separation attacks.

Previous work on LISP has mainly tackled the evaluation of the performance of LISP [8], [9], neglecting addressing security issues. Some work has also focused on the use of LISP as actually a mechanism to mitigate some non-LISP related DDoS attacks [10], [11] Some proposals focused specifically on the LISP data plane confidentiality [12], [13], extending LISP to allow protecting data exchange.

Concerning the LISP MDS, there has been quite some work on how to secure it to provide origin authentication and secure communication [14], [15], or integrity and anti-replay protection [16]. Saucez et al. [17], explored the security threats applicable to the whole LISP architecture, also providing recommendations to make the whole LISP infrastructure more robust. They are the first to conjecture the possibility of an amplification attack leveraging on the MDS. Nevertheless, except for a basic calculation of theoretical amplification factor, no analysis whatsoever has been carried out, no details are provided on how to realize the attack in the context of LISP–DDT, and proof-of-concept attack has been carried out, which are the main contributions of this paper.

The remainder of this paper is organized as follows: Sec. II provides the necessary background on LISP; Sec. III provides a proof-of-concept of the attack based on GNS-3; finally, Sec. IV concludes this paper by summarizing its main achievements.

## II. LISP BACKGROUND

In order to provide an overview on LISP, we first show how the LISP data plane (i.e., packet forwarding) works (Sec. II-1), before describing the LISP control plane (Sec. II-2).

*1) LISP Data Plane:* The *Locator/IDentifier Separation Protocol* (LISP) [7] separates the identification and localization roles of IP addresses by introducing two logical addressing spaces: ($i$) the *Routing LOCator space* (RLOC), which is globally routable; ($ii$) the *Endpoint IDentifier space* (EID), which is only locally routable and whose main purpose is to identify the communication endpoint. With this separation, the Internet core, also known as *Default Free Zone* (DFZ), handles RLOCs addresses like it is done today, i.e., maintaining routes so that packets can be forwarded between any router within the DFZ. Stub networks instead use the EID addressing space.

The implication of such a separation lies in stub networks not needing anymore a full knowledge of the Internet routing information, whereby the DFZ does not need anymore to advertise the EID space in its routing infrastructure. Nonetheless, in order to provide end-to-end communication, another level of indirection is required.

The LISP data plane provides this level of indirection through a tunneling mechanism over the DFZ. More specifically, any communicating host generates regular IP packets using its EID as the source address and the destination EID as the destination address. Forwarding towards the border router is done as usual in the local domain. The border router, now called *Ingress Tunnel Router* (ITR), will encapsulate the packets using the RLOC addressing space, i.e., using its RLOC address as the source address and the destination RLOC as the destination address in the tunnel header encapsulating the original packet [7]. The encapsulated packets can now be forwarded over the DFZ. The border router at the destination site, now called *Egress Tunnel Router* (ETR), will decapsulate the LISP packets so that the original packet can be forwarded to its final EID destination.

*2) LISP Control Plane:* In order to perform the data plane operations, tunnel routers need to be able to associate EIDs to RLOCs. The binding between the two addressing spaces is named *mapping*. A mapping enables a tunnel router (generally referred to as an *xTR*) to retrieve the RLOCs associated to a given EID, to be used in the outer header when encapsulating. Mappings are stored in two data structures present on xTRs: $(i)$ the LISP *Database* storing the mappings for EID prefixes for which the xTR itself is an RLOC; $(ii)$ the LISP *Cache* storing mappings for EID prefixes used in ongoing communication towards/from distant LISP sites.

In addition, the LISP control plane relies on a pull model, i.e., pulling routing information on xTRs only when actually needed. The key point of this approach is *how to make routing information available on an on-demand fashion?* To this end, the LISP control plane introduces a *Mapping Distribution System (*MDS*)* providing a lookup infrastructure from where mappings can be retrieved upon an explicit query.

From an abstract point of view, the MDS works as follows. The ITR that needs a mapping for a new flow first sends a query, consisting of a `Map-Request` message to a *Map-Resolver* (MR) [18]. The query is forwarded by the Map-Resolver inside the MDS according to the specific protocol/architecture used, to reach the *Map-Server* (MS) where the site using the requested EID has registered the mapping. The Map-Server then forwards the query to the xTR that registered such a mapping. In turn, the xTR will send the reply, consisting of a `Map-Reply` message containing the requested mapping, directly to the ITR that, in the first place, sent the query. The Map-Resolver and Map-Server elements represent respectively where to ask for a mapping and where to register a mapping so as to make it available to other LISP sites. They provide a general front-end for any mapping system, *"hiding"* the specific MDS in use to the LISP tunnel routers.

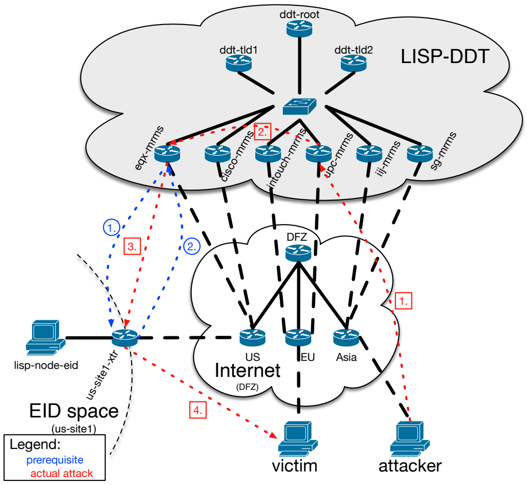


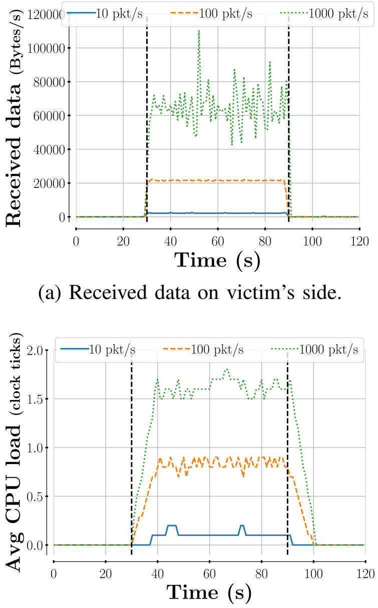
Fig. 1. GNS-3 testbed and illustration of the steps of the amplification attack.

Currently, LISP *Delegated Database Tree* (LISP–DDT [19]) is the MDS used in the experimental LISP Beta Network [20] and, as such, is the subject of this paper. LISP–DDT is a DNS-like system with a hierarchy of LISP–DDT Servers, as illustrated in the top part of Fig. 1. When a mapping must be retrieved for a given EID, a root server (`ddt-root` in Fig. 1) is queried first. By definition, a root server is responsible for the entire EID space, which is divided into several portions, each one being managed by one of the Top-Level Domain (`ddt-tld*` in Fig. 1). The root replies with a pointer (i.e., a referral) to its children responsible for the EID prefix to resolve. The process is recursively repeated with the returned child considered as the root of the sub-tree, where a mapping for the EID can be retrieved. This recursive process is stopped when a leaf has been reached. In LISP–DDT, leaves are made of Map-Servers (`*-mrms` in Fig. 1). Each Map-Server maintains a list of ETR authoritative for the different EID prefixes registered to it (at least one matching the requested EID). Thus, when the leaf has been reached, the mapping is retrieved by sending a `Map-Request` to one of the ETR authoritative for the matching EID prefix. The ETR, will generate the corresponding `Map-Reply`, containing the requested mapping, and send it directly to the ITR who originally sent out the `Map-Request`.

## III. LISP & AMPLIFICATION ATTACKS

In the context of LISP, a Denial-of-Service (DoS) attack works in a similar way in DNS: the attacker forges `Map-Request`, with the source address being spoofed with the victim address (this is possible, as lots of ISPs still allow IP spoofing [21]). The MDS will reply with `Map-Reply` messages destined to the victim. If `Map-Reply` packets are large enough, it has the potential to drown the victim (and possibly part of the victim LISP network).

We actually realize the amplification attack's proof-of-concept, relying on the LISP MDS. Simulations are run based



(a) Received data on victim's side.

(b) CPU overload on victim's side.

Fig. 2. Effects of the DoS attack on the victim.

on GNS-3, an emulator in a virtualized router.[1] Our simulations are based on the LISP Beta Network [20]. Since March 2012, the LISP Beta Network uses LISP–DDT as Mapping System [22]. Participants in the LISP Beta Network are located in 27 different countries, mostly in Europe and USA, and consist of both academic institutions and companies. The LISP–DDT hierarchy of the testbed is composed of nine servers composing a three-level topology. The root (`ddt-root`) is made of one server running in the USA and reachable via anycast addresses. The hierarchy is composed of only one level below the root, composed of two Top-Level Domains (`ddt-tld⋆`), also reachable in anycast. Below them, there are six MS/MR servers (`⋆-mrms`) implementing the front-end through which the MDS can be queried. We replicated such topology with GNS-3, as shown in Fig. 1, where their name can clearly identify servers' roles. To emulate the deployment mentioned above in our physical testbed, we used four servers organized in two subnets with two servers each. Both subnets are connected through a link with a bottleneck of 4Mbits/sec. One of the servers in the first subnet is in charge of the MDS (LISP–DDT on Fig. 1), implemented with GNS-3 running Cisco IOS C7200 image.[2] One of the servers in the second subnet is also running GNS-3 and simulates routers in the Internet (DFZ, US, . . . on Fig. 1), as well as the EID space. The remaining servers are dedicated, respectively, to the attacker and the victim and are running as dockers on Debian.

*1) Amplification Attack in* LISP*:* The attack itself works in four steps, represented by red arrows in Fig. 1:

**Step 1:** The attacker crafts `Map-Request` messages for the EID prefix of `us-site1-xtr`. Those messages rely on spoofing, which is possible due to insecure operators [21]. In the Outer header, the attacker will specify, as the destination, any available Map-Server (here `upc-mrms`) and its own IP address as the source.

**Step 2:** The forged `Map-Request` is sent to a Map-Server (`upc-mrms` in Fig. 1). The `Map-Request` will be forwarded to a Map-Server associated with `us-site1` (`eqx-mrms` or `cisco-mrms` in Fig. 1) through the LISP–DDT hierarchy.

**Step 3:** The Map-Server forwards the `Map-Request` to one of the authoritative ETR in `us-site1` (`us-site1-xtr1` in Fig. 1).

**Step 4:** Finally, the ETR (`us-site1-xtr1`) sends back the `Map-Reply`, larger than the `Map-Request`, to the victim. The attack is possible only if the following two conditions are met (blue arrows on Fig. 1): (*i*) the site used as the vector for the attack must have registered a potentially long list of RLOCs in the LISP MDS; (*ii*) the LISP MDS must have acknowledged the registration. We argue that such conditions are easily reached if LISP is deployed at Internet scale because of the rich connectivity among AS, however, this is left as future work.

*2) Results:* In our testbed, we have configured `us-site1-xtr` to register four IPv4 RLOCs (including itself) and four IPv6 RLOCs, leading to a `Map-Reply` of 214 bytes. The fact that registered RLOCs do not exist (except `us-site1-xtr`) is not the problem: they will be part of the `Map-Reply` with the flag Unreachable set. `Map-Request` sent by the attacker are 102 Bytes long, leading to an amplification factor of 2.09. A Python server has been implemented on the victim for confirming the data reception. The attacker forges crafted packets and launches them at a given rate using a Python code developed for querying the LISP MDS [23].[3] Fig. 2 shows the effects of the DoS attack on the victim. Our simulations last 90 seconds and are divided into three parts: [0;30s] corresponds to the period before the attack, [31s;90s] to the attack itself, and, finally, [91s;120s] to the period after the attack. Those three periods are delimited by a vertical dashed bar in Fig. 2. We performed the attack at three levels of intensity: 10 `Map-Request` per second (˜1KB/sec – blue line), 100 `Map-Request` per second (˜10KB/sec – orange line), and 1,000 `Map-Request` per second (˜102KB/sec – green line), that corresponds to the maximum allowed by our testbed). Fig. 2a shows the amount of data received, by the victim, during the attack, respectively ˜2KB/sec, ˜21KB/sec and ˜64KB/sec in our three scenarios. We notice that, as soon as the attack has started, the amount of received data increases rapidly, quickly reaching its maximum. It is worth noticing that, for low rates, the victim receives all packets (corresponding to the theoretical amplification factor). In contrast, for 1,000 packets/sec, packets were lost due to our testbed's physical limits, leading to the oscillations observed in Fig. 2a. This figure clearly shows how compared to the traffic generated by the attacker

[1] See https://gns3.com/

[2] We run GNS-3 on Virtual PC Simulator, a lightweight operating system.

[3] All our scripts (GNS-3 topology, victim, attack) are freely available at: https://gitlab.com/m.gabriel/lisp-ms-ddos

(see above), the volume of traffic received by the victim is double. Fig. 2b shows the CPU load on the victim, caused by the attack. More precisely, the CPU load of our Python script (running on the victim server) during the simulation, computed as an average, moving every second, over a time window of the last ten seconds. Fig. 2b clearly shows how CPU load increases with the attack intensity (the higher the attack, the higher the CPU load). All in all, Fig. 2 proofs how an attacker can create a DoS attack by consuming victim's bandwidth (Fig. 2a) and wasting victim's CPU clock cycles (Fig. 2b).

*3) Possible Mitigation Techniques:* On the one hand, because at the very end the attack we showcase here remains a DoS attack with spoofing, any mechanism able to detect DoS attacks (e.g. [24]) and spoofing (e.g. [21], [25]) can be used. On the other hand, LISP implementations can be made more robust. The first and foremost protection is to enforce security checks during `Map-Request` processing. More specifically, because of the peculiar encapsulation used by LISP, only the source address of the inner header `Map-Request` packets has to be spoofed. At the same time, the outer one does not need to be spoofed (making anti-spoofing solutions less effective). In this way, an attacker can bypass source address filtering very easily. However, Map-Resolvers have the possibility to understand whether there is something wrong going on. The simplest thing to do is to check whether inner and outer source addresses are the same. If they are, either the request is legitimate or both addresses are spoofed, but in the latter case, the Map-Resolver has no way to know it. Note that the LISP specification allows using of EID in the inner header and RLOCs in the outer header, which leads to address mismatch. However, in this case, a mapping must exist, proving that actually, the inner header EID can be mapped to the outer header RLOC. Certainly, retrieving the corresponding mapping will generate a performance penalty; however, it will also strongly reduce the possibility of performing the attack.

## IV. Conclusion

The increasing interest in the locator/identifier separation paradigm to boost the scalability of the Internet while providing undeniable benefits may as well open the possibility of various forms of attacks. Focusing on the LISP Mapping Distribution System we showcase how it can be abused to carry out amplification attacks. We built GNS-3 simulated testbed, mimicking the real LISP Beta Network, showing how an attack can be carried out. We were able to show how an attacker can consume two of the primary resources of a victim, namely network bandwidth and CPU cycles.

An interesting future work will be to explore the size of the amplification attack if locator/identifier separation is deployed at Internet scale. Indeed, our proof of concept actually consumes a very small amount of bandwidth, and does not allow to assess the real impact at large scale. Also, in the LISP specific case, it would be worth exploring what kind of mechanism could be added to the protocol to make it less easily used as an attack vector.

## References

[1] D. Meyer, L. Zhang, and K. Fall, "Report from the IAB workshop on routing and addressing," Internet Engineering Task Force, RFC 4984, September 2007.

[2] "Focus Group on Technologies for Network 2030," see https://www.itu.int/en/ITU-T/focusgroups/net2030/Pages/default.aspx.

[3] A. Yastrebova, R. Kirichek, Y. Koucheryavy, A. Borodin, and A. Koucheryavy, "Future networks 2030: Architecture & requirements," in *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*. IEEE, 2018, pp. 1–8.

[4] B. Quoitin, L. Iannone, C. De Launois, and O. Bonaventure, "Evaluating the benefits of the locator/identifier separation," in *Proc. ACM/IEEE International Workshop on Mobility in the Evolving Internet Architecture (MobiArch)*, August 2007.

[5] T. Li, "Recommendation for a routing architecture," Internet Research Task Force, RFC 6115, February 2011.

[6] G. Kambourakis, A. Moshos, D. Geneiatakis, and S. Gritzalis, "Detecting DNS amplification attacks," in *Proc. International Workshop on Critical Information Infrastructures Security*, October 2007.

[7] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "Locator/ID separation protocol (LISP)," Internet Engineering Task Force, RFC 6830, Jan. 2013.

[8] L. Iannone and O. Bonaventure, "On the cost of caching Locator/ID mappings," in *Proc. ACM SIGCOMM CoNEXT*, December 2007.

[9] H. Zhang, M. Chen, and Y. Zhu, "Evaluating the performance on IDLoc mapping," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, November 2008.

[10] K. Okada, H. Hazeyama, and Y. Kadobayashi, "Oblivious DDoS mitigation with Locator/ID separation protocol," in *Proc. International Conference on Future Internet Technologies (CFI)*, June 2014.

[11] R. P. Martins, J. L. Martis, and H. J. L. Domingos, "EIP – Preventing DDoS with Ephemeral IP Identifiers Cryptographically Generated," arXiv, coRR abs/1612.07065, December 2016.

[12] D. Farinacci and B. Weis, "Locator/id separation protocol (LISP) data-plane confidentiality," Internet Engineering Task Force, RFC 8061, February 2017.

[13] M. M. Kallash, J. Loo, A. Lasebae, and M. Aiash, "A security framework for node-to-node communications based on the LISP architecture," *International Journal of Electronic Security and Digital Forensics*, vol. 10, no. 3, pp. 276–291, January 2018.

[14] V. P. Kafle, R. Li, D. Inoue, and H. Harai, "An integrated security scheme for ID/Locator split architecture of future ?etwork," in *Proc. IEEE International Conference on Communications (ICC)*, June 2012.

[15] M. Aiash, A. Al-Nemrat, and D. Preston, "Securing address registration in Location/ID split protocol using ID-based cryptography," in *Proc. Wired/Wireless Internet Communication (WWIC)*, June 2013.

[16] F. Maino, V. Ermagan, C. Albert, and D. Saucez, "LISP-security (LISP-SEC)," Internet Engineering Task Force, Internet Draft (Work in Progress) draft-ietf-lisp-sec-20, January 2020.

[17] D. Saucez, L. Iannone, and O. Bonaventure, *The Map-and-Encap Locator/Identifier Separation Paradigm: a Security Analysis*, ser. Advances in Web Technologies and Engineering. IGI Global, 2013, pp. 148–163.

[18] V. Fuller and D. Farinacci, "Locator/ID separation protocol (LISP) map-server interface," Internet Engineering Task Force, RFC 6833, Jan. 2013.

[19] V. Fuller, D. Lewis, V. Ermagan, A. Jain, and A. Smirnov, "Locator/ID separation protocol delegated database tree (LISP-DDT)," Internet Engineering Task Force, RFC 8111, May 2017.

[20] "LISP Beta Network," see http://www.lisp4.net.

[21] R. Beverly, A. Berger, Y. Hyun, and k. claffy, "Understanding the efficacy of deployed Internet source address validation filtering," in *Proc. ACM Internet Measurement Conference (IMC)*, November 2010.

[22] D. Saucez, L. Iannone, and B. Donnet, "A first measurement look at the deployment and evolution of the Locator/ID separation protocol," *ACM SIGCOMM CCR*, vol. 43, no. 1, pp. 37–43, April 2013.

[23] Y. Li, A. Abouseif, L. Iannone, and D. Saucez, "LISP-Views: Monitoring LISP at large scale," in *Proc. International Teletraffic Congress (ITC)*, September 2017.

[24] R. Doriguzzi-Corin, S. Millar, S. Scott-Hayward, J. Martinez-del Rincon, and D. Siracusa, "LUCID: A pratical, lightweight deep learning solution for DDoS attack detection," *IEEE Transactions on Network and Service Management (TNSM)*, February 2020.

[25] P. Ferguson and D. Senie, "Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing," Internet Engineering Task Force, RFC 2827, May 2000.