

## An Architectural Introduction to the Locator/ID Separation Protocol (LISP)

Архитектурное введение в протокол LISP

### Аннотация

Этот документ описывает архитектуру протокола разделения идентификаторов и локаторов (Locator/ID Separation Protocol или LISP), упрощая понимание спецификаций LISP и обеспечивая базу для обсуждения деталей протоколов LISP. Документ служит введением, а подробные сведения приведены в спецификациях RFC 9300 и RFC 9301.

### Статус документа

Документ не содержит стандарта Internet и публикуется с информационными целями.

Документ является результатом работы IETF<sup>1</sup> и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG<sup>2</sup>. Не все документы, одобренные IESG претендуют на статус Internet Standard. Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 7841.

Информация о текущем статусе документа, найденных ошибках и способах обратной связи доступна по ссылке <https://www.rfc-editor.org/info/rfc9299>.

### Авторские права

Copyright (c) 2022. Авторские права принадлежат IETF Trust и лицам, указанным в качестве авторов документа. Все права защищены.

К документу применимы права и ограничения, указанные в BCP 78 и IETF Trust Legal Provisions и относящиеся к документам IETF (<http://trustee.ietf.org/license-info>), на момент публикации данного документа. Прочтите упомянутые документы внимательно. Фрагменты программного кода, включённые в этот документ, распространяются в соответствии с упрощённой лицензией BSD, как указано в параграфе 4.e документа IETF Trust Legal Provisions, без каких-либо гарантий (как указано в Simplified BSD License).

## Оглавление

1. Введение.....	2
2. Определения терминов.....	2
3. Архитектура LISP.....	2
3.1. Принципы устройства протокола.....	3
3.2. Обзор архитектуры.....	3
3.3. Плоскость данных.....	4
3.3.1. Инкапсуляция LISP.....	4
3.3.2. Состояние пересылки LISP.....	5
3.4. Плоскость управления.....	5
3.4.1. Сопоставления LISP.....	5
3.4.2. Интерфейс системы сопоставления.....	5
3.4.3. Система сопоставления.....	5
3.4.3.1. LISP-ALT.....	6
3.4.3.2. LISP-DDT.....	6
3.5. Механизмы межсетевого взаимодействия.....	6
4. Механизмы работы LISP.....	7
4.1. Поддержка кэша.....	7
4.2. Достижимость RLOC.....	7
4.3. Синхронизация ETR.....	8
4.4. Обработка MTU.....	8
5. Мобильность.....	8
6. Групповая передача.....	8
7. Варианты применения.....	9
7.1. Организация трафика.....	9
7.2. LISP для сосуществования с IPv6.....	9
7.3. LISP для VPN.....	9
7.4. LISP для мобильности VM в ЦОД.....	9
8. Вопросы безопасности.....	9
9. Взаимодействие с IANA.....	10
10. Литература.....	10
10.1. Нормативные документы.....	10
10.2. Дополнительная литература.....	11
Приложение А. История разделения идентификаторов и локаторов.....	11
А.1. Старые модели LISP.....	11

<sup>1</sup>Internet Engineering Task Force - комиссия по решению инженерных задач Internet.

<sup>2</sup>Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

Благодарности.....	12
Адреса авторов.....	12

## 1. Введение

Этот документ описывает архитектуру протокола разделения идентификаторов и локаторов (LISP) [RFC9300] [RFC9301], основные механизмы работы и принципы устройства. По сути, LISP основан на хорошо известной архитектурной идее освобождения от перегрузки семантики адресов IP. Как отметил Noel Chiappa [RFC4984], в настоящее время адреса IP указывают сразу топологическое местоположение точки подключения к сети и идентификацию узлов. Однако требования узлов и маршрутизации принципиально различаются. Системам маршрутизации нужна агрегируемость адресов и их топологическая значимость, а узлам требуется идентификация, независимая от конкретного местоположения [RFC4984].

LISP создаёт два отдельных пространства имён для идентификаторов конечных точек (Endpoint Identifier или EID) и локаторов маршрутизации (Routing Locator или RLOC). Оба пространства синтаксически идентичны современным адресам IPv4 и IPv6. Однако EID служат для однозначной идентификации узлов независимо от их топологического местоположения и обычно маршрутизируются внутри домена. Локаторы RLOC назначаются топологически и обычно маршрутизируются между доменами. С помощью LISP можно логически разделить границу Internet (места подключения узлов) и ядро (где происходит междоменная маршрутизация). Маршрутизаторы с поддержкой LISP соединяют эти логические пространства. LISP поддерживает базу данных, называемую системой сопоставления (Mapping System), для хранения и извлечения сведений о сопоставлении отождествлений и местоположений. Маршрутизаторы с поддержкой LISP обмениваются пакетами через ядро Internet, инкапсулируя их в нужное место.

- Локаторы RLOC имеют смысл лишь в базовой (underlay) сети, т. е. в ядре маршрутизации.
- EID имеют смысл лишь в наложенной сети, которая является связующей инкапсуляцией между маршрутизаторами с поддержкой LISP.
- На границе LISP идентификаторы EID сопоставляются с RLOC.
- В базовой сети RLOC служат сразу локаторами и идентификаторами.
- EID внутри сайта LISP служат идентификаторами и локаторами для других узлов этого сайта.
- EID внутри сайта LISP передаёт идентификатор и ограниченную семантику локатора узлам других сайтов LISP (т. е. достаточно локатора, чтобы понять, что EID является внешним для сайта).

Указанные выше свойства не уникальны для LISP и присущи многим наложенным протоколам.

Исходную мотивацию разработки LISP можно найти в проблеме расширяемости маршрутизации [RFC4984], где при полном развёртывании LISP ядро Internet будет заполнено RLOC, а механизмы организации трафика (Traffic Engineering или TE) будут вынесены в систему сопоставления. В таком варианте локаторы RLOC будут квазистатическими (т. е. меняющимися редко), что делает систему маршрутизации расширяемой [Quoitin], а EID могут перемещаться свободно, «не создавая пены» в базовой системе глобальной маршрутизации. В [RFC7215] рассмотрено влияние LISP на систему глобальной маршрутизации в процессе перехода. Однако разделение местоположения и идентификации, предлагаемое LISP, подходит и для применения в других случаях, таких как TE, многодомные подключения и мобильность узлов.

В этом документе описана архитектура и основные механизмы работы LISP, а также принципы разработки. Важно отметить, что документ не задаёт и не дополняет LISP. Заинтересованным читателям следует обратиться к спецификациям LISP ([RFC9300] и [RFC9301]), а также дополнительным документам ([RFC6831], [RFC6832], [RFC9302], [RFC6835], [RFC6836], [RFC7052]) для спецификации протокола и рекомендаций по развёртыванию LISP [RFC7215].

## 2. Определения терминов

### **Endpoint Identifier (EID) - идентификатор конечной точки**

Адрес, служащий для однозначного указания узла независимо от топологического размещения. Обычно маршрутизируется внутри домена.

### **Routing Locator (RLOC) - локатор маршрутизации**

Адрес, топологически назначаемый точке подключения к сети. Обычно маршрутизируется между доменами.

### **Ingress Tunnel Router (ITR) - входной маршрутизатор туннеля**

Поддерживающий LISP маршрутизатор, инкапсулирующий пакеты от сайта LISP в направлении ядра сети.

### **Egress Tunnel Router (ETR) - выходной маршрутизатор туннеля**

Поддерживающий LISP маршрутизатор, деинкапсулирующий пакеты из ядра сети в направлении сайта LISP.

### **xTR**

Маршрутизатор, реализующий функции ITR и ETR.

### **Map-Request**

Сигнальное сообщение LISP, служащее для запроса отображения EID на RLOC.

### **Map-Reply**

Сигнальное сообщение LISP, служащее ответом на Map-Request и содержащее нужное отображение EID на RLOC.

### **Map-Register**

Сигнальное сообщение LISP, служащее для регистрации отображения EID на RLOC.

### **Map-Notify**

Сигнальное сообщение LISP, передаваемое в ответ на Map-Register для подтверждения корректного получения отображения EID на RLOC.

Этот документ описывает архитектуру LISP и не вносит новых терминов. Определения используемых терминов даны в [RFC9300], [RFC9301], [RFC6831], [RFC6832], [RFC9302], [RFC6835], [RFC6836], [RFC7052], [RFC7215].

## 3. Архитектура LISP

В этом разделе представлена архитектура LISP. Сначала рассмотрены принципы устройства LISP, затем более подробно описаны основные аспекты: плоскости данных и управления, механизмы межсетевое взаимодействия.

### 3.1. Принципы устройства протокола

Архитектура LISP основана на 4 базовых принципах.

#### Разделение локаторов и идентификаторов

Разделение перегруженной сегодня семантики адресов IP позволяет устройствам иметь основанный на отождествлении (идентификаторе) адрес, отвязанный от топологически значимого адреса. Предоставление доступа в ядро Internet лишь топологически значимым адресам позволяет агрегировать их для поддержки расширяемости маршрутизации. Отдельным устройствам назначаются основанные на отождествлениях адреса, не применяемые для пересылки в ядре Internet.

#### Наложение

Эта архитектура накладывает маршрутизацию пакетов на текущую инфраструктуру Internet, позволяя развёртывать новые протоколы без изменения этой инфраструктуры, что существенно снижает расходы.

#### Отделение плоскости данных от плоскости управления

Разделение плоскостей данных и управления позволяет расширять их независимо и применять в них разные архитектурные подходы. Это важно с учётом разных требований плоскостей и позволяет добавлять другие плоскости. Несмотря на разделение плоскостей данных и управления, они не изолированы полностью, поскольку плоскость данных LISP может вызывать действия в плоскости управления.

#### Возможность поэтапного развёртывания

Этот принцип обеспечивает совместимость протокола с унаследованной инфраструктурой Internet, сохраняя целевые преимущества для ранних пользователей.

### 3.2. Обзор архитектуры

LISP архитектурно отделяет ядро от периметра Internet, создавая два пространства имён - EID и RLOC. Периметр образуют сайты LISP (например, автономные системы - AS), использующие адреса EID. Идентификаторы EID - это адреса IPv4 или IPv6, однозначно указывающие конечные хосты, которые назначаются и настраиваются с использованием механизмов, существовавших на момент разработки протокола. В EID не включаются междоменные топологические данные и поэтому EID обычно маршрутизируются на периферии (внутри сайта LISP), но не в ядре. Взаимодействие сайтов LISP с сайтами и доменами без поддержки LISP в сети Internet описано в параграфе 3.5. Механизмы межсетевое взаимодействия.

Сайты LISP (на периметре) подключаются к межсетевому ядру Internet через поддерживающие LISP маршрутизаторы (например, граничные маршрутизаторы). Сайты LISP соединяются через межсетевое ядро Internet с использованием туннелей между поддерживающими LISP маршрутизаторами. Когда пакет от сайта LISP направлен в сеть ядра, он попадает в инкапсулированный туннель через входной маршрутизатор ITR. Пакеты из ядра сети на сайт LISP выходят из инкапсулированного туннеля через выходной маршрутизатор ETR. XTR - это маршрутизатор, выполняющий функции ITR и ETR. В этом контексте ITR инкапсулируют пакеты, а ETR деинкапсулируют их, следовательно LISP работает как наложение на имеющееся ядро Internet.

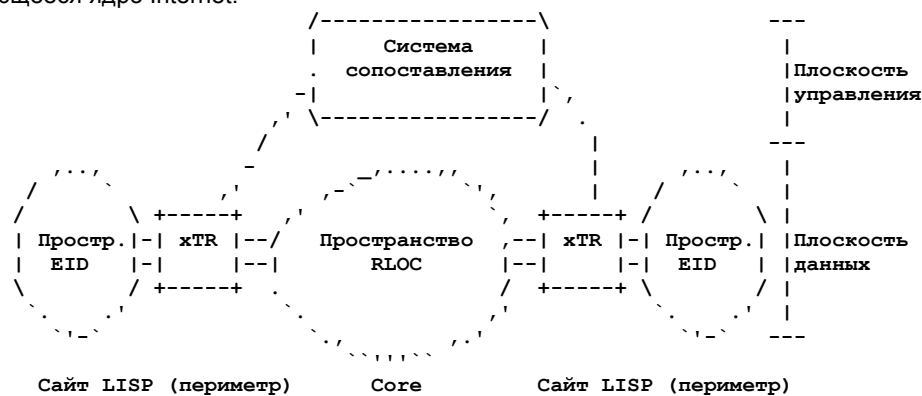


Рисунок 1. Схема архитектуры LISP.

При использовании LISP ядро работает с локаторами RLOC. Локатор RLOC - это адрес IPv4 или IPv6, назначенный интерфейсу маршрутизатора ITR или ETR в сторону ядра.

База данных (обычно распределенная), которую называют системой сопоставления (Mapping System), содержит сопоставления между EID и RLOC. Такие сопоставления связывают идентификаторы устройств, подключённых к сайту LISP (EID), с набором RLOC, настроенных на поддерживающих LISP маршрутизаторах, обслуживающих сайт. Кроме того, сопоставления включают правила TE и могут быть настроены на поддержку многодомности и распределения нагрузки. Система сопоставления LISP концептуально похожа на DNS и организована как распределенная по сети база данных множества организаций. В LISP маршрутизаторы ETR регистрируют сопоставления, а ITR извлекают их.

В архитектуре LISP уделено особое внимание поэтапному развёртыванию. С учётом того, что LISP представляет наложение на текущую архитектуру Internet, конечные хосты, а также внутридомуемые и междоменные маршрутизаторы не требуют изменений. В существующей инфраструктуре требуется изменять лишь маршрутизаторы, соединяющие между собой пространства EID и RLOC. Кроме того, LISP требует развёртывания независимой системы сопоставления, такая распределенная база данных является новым элементом сети.

Ниже (Рисунок 2) представлена упрощённая последовательность потока пакетов между парой узлов, подключённых к сайтам LISP. Отметим, что типичные маршрутизаторы с поддержкой LISP - это xTR (ITR и ETR). Клиент HostA на рисунке передаёт пакет серверу HostB.



Рисунок 2. Последовательность потока пакетов в LISP.

1. HostA извлекает EID\_B для HostB, обычно запрашивая DNS и получая запись A или AAAA. Затем он создаёт пакет IP как в Internet. Пакет имеет адрес отправителя EID\_A и адрес получателя EID\_B.
2. Пакет пересылается в ITR\_A сайта LISP с использованием стандартных внутридоменных механизмов.
3. ITR\_A при получении пакета запрашивает систему сопоставления для получения локатора ETR\_B, обслуживающего EID\_B сервера HostB. Для этого служит управляющее сообщение LISP, называемое Map-Request. Сообщение содержит EID\_B в качестве ключа поиска. В ответ ITR\_A получает другое управляющее сообщение LISP - Map-Reply. Это сообщение содержит два локатора RLOC\_B1 и RLOC\_B2, а также правила TE - приоритет и вес для каждого локатора. Отметим, что при необходимости Map-Reply может включать большее число локаторов. ITR\_A может кэшировать сопоставления в локальном хранилище для ускорения пересылки последующих пакетов.
4. ITR\_A инкапсулирует пакет в сторону RLOC\_B1 (выбирается в соответствии с приоритетом и весом в сопоставлении). Пакет имеет два заголовка IP. Внешний заголовок указывает RLOC\_A1 как источник и RLOC\_B1 - как получателя. Во внутреннем исходном заголовке пакета указан адрес источника EID\_A и получателя EID\_B. Затем ITR\_A добавляет заголовок LISP. Более подробно процесс описан в параграфе 3.3.1. Инкапсуляция LISP.
5. Инкапсулированный пакет пересылается через межсетевое ядро как обычный пакет IP, оставляя EID невидимым для ядра.
6. При получении инкапсулированного пакета маршрутизатор ETR\_B деинкапсулирует его и пересылает в HostB.

### 3.3. Плоскость данных

В этом параграфе приведено высокоуровневое описание плоскости данных LISP, детально заданной в [RFC9300]. Плоскость данных LISP отвечает за инкапсуляцию и деинкапсуляцию пакетов данных и кэширование соответствующего состояния пересылки. Основными элементами плоскости данных являются маршрутизаторы ITR и ETR. Оба типа маршрутизаторов поддерживают LISP и соединяют EID с пространством RLOC (ITR) и наоборот (ETR).

#### 3.3.1. Инкапсуляция LISP

Маршрутизаторы ITR инкапсулируют пакеты данных в направлении ETR. Пакеты данных LISP инкапсулируются с использованием UDP (порт 4341). Порт отправителя ITR обычно выбирает с использованием хэш-значения квинтета (5-tuple) из внутреннего заголовка (для согласованности при использовании нескольких путей, как в ECMP [RFC2992]) и игнорируется при получении. Пакеты данных LISP часто инкапсулируются в пакеты UDP с нулевой контрольной суммой [RFC6935] [RFC6936], которую нельзя проверить при получении, поскольку внутри пакета LISP имеется транспортный заголовок с проверяемой контрольной суммой. Использование нулевой контрольной суммы UDP при передаче по протоколу IPv6 для всех протоколов туннелирования, подобных LISP, регулируется заявлением о применимости из [RFC6936]. Если пакеты данных LISP инкапсулируются в пакеты UDP с ненулевой контрольной суммой, внешняя контрольная сумма проверяется при получении пакетов UDP как при обычной обработке UDP.

Пакеты с инкапсуляцией LISP включают также заголовок LISP (между внешним заголовком UDP и исходным заголовком IP). Заголовок LISP устанавливают маршрутизаторы ITR и вырезают ETR. Заголовок содержит сведения о достижимости (4.2. Достижимость RLOC) и поле Instance ID, служащее для разделения трафика разных арендаторов на сайте LISP, что позволяет применять на сайтах перекрывающуюся, но логически разделенную адресацию EID.

В результате LISP работает с 4 заголовками - внутренним заголовком от источника и 3 заголовками, которые LISP добавляет перед ним (от внешнего к внутреннему), как указано ниже.

1. Внешний заголовок IP с RLOC в качестве адресов отправителя и получателя. Этот заголовок создаёт маршрутизатор ITR и вырезает маршрутизатор ETR.
2. Заголовок UDP (порт 4341), обычно с нулевой контрольной суммой, создаваемый ITR и вырезаемый ETR.
3. Заголовок LISP со свойствами плоскости пересылки (такими как доступность) и полем Instance ID. Этот заголовок создаёт маршрутизатор ITR и вырезает маршрутизатор ETR.
4. Внутренний заголовок IP с EID в качестве адресов отправителя и получателя. Этот заголовок создаёт хост-источник и он сохраняется неизменным при обработке в плоскости данных LISP на ITR и ETR.



В некоторых случаях полезна повторная инкапсуляция и/или рекурсивные туннели для выбора конкретного пути через базовую сеть, например, для предотвращения перегрузки или отказа. Туннелирование с повторной инкапсуляцией является последовательным применением туннелей LISP и выполняется при удалении декапсулятором (ETR) заголовка LISP и новой инкапсуляции (ITR) с добавлением другого заголовка. Рекурсивные туннели являются вложенными и реализуются с использованием для пакета неоднократной инкапсуляции LISP. Такие функции реализуются маршрутизаторами с реинкапсуляцией туннелей (Re-encapsulating Tunnel Router или RTR). Маршрутизатор RTR можно считать выступающим сначала в роли ETR для декапсуляции пакетов, затем - в роли ITR с инкапсуляцией пакетов для другого локатора (см. [RFC9300] и [RFC9301]).

### 3.3.2. Состояние пересылки LISP

В архитектуре LISP маршрутизаторы ITR сохраняют лишь сведения, достаточные для маршрутизации проходящего через них трафика. Иными словами, ITR нужно лишь извлечь из системы сопоставления LISP отображения EID-Prefix (блок EID) на RLOC, применяемые для инкапсуляции пакетов. Такие сопоставления хранятся в локальном кэше, называемом LISP Map-Cache, для последующих пакетов, направляемых в тот же EID-Prefix. Отметим, что в случае перекрытия EID-Prefix маршрутизатор ITR может получить по запросу набор сопоставлений из запрошенного EID-Prefix и более конкретных EID-Prefix (см. параграф 5.5 в [RFC9301]). Отображения включают значения TTL, устанавливаемые ETR. Дополнительные сведения об управлении Map-Cache приведены в параграфе 4.1. Поддержка кэша.

## 3.4. Плоскость управления

Плоскость управления LISP, заданная в [RFC9301], обеспечивает стандартный интерфейс для регистрации и запроса сопоставлений. Система сопоставления LISP - это база данных, хранящая отображения. В последующих параграфах сначала описаны сами отображения, а затем стандартный интерфейс системы сопоставления и её архитектура.

### 3.4.1. Сопоставления LISP

Каждое отображение включает привязки EID-Prefix к набору RLOC, а также правила TE в виде приоритета и веса для RLOC. Приоритет позволяет ETR настроить активную и резервную политику, а вес служит для распределения нагрузки (трафика) между RLOC (по потокам).

Типичное сопоставление в LISP связывает EID в форме префиксов IP с набором RLOC тоже в форме адресов IP. Адреса IPv4 и IPv6 кодируются с использованием подходящего идентификатора семейства адресов (Address Family Identifier или AFI) [RFC8060]. Однако LISP может поддерживать более общее кодирование адресов на канонического формате LISP (LISP Canonical Address Format или LCAF) [RFC8060]. С помощью такого обобщённого синтаксиса кодирования адресов LISP стремится обеспечить гибкость имеющихся и будущих приложений. Например, LCAF позволяет поддерживать адреса MAC (Media Access Control), географические координаты, имена ASCII и задаваемые приложением данные.

### 3.4.2. Интерфейс системы сопоставления

LISP задаёт стандартный интерфейс между плоскостями данных и управления в [RFC9301], включающий 2 сущности.

#### **Map-Server - сервер сопоставлений**

Компонент сетевой инфраструктуры, изучающий отображения через маршрутизаторы ETR и публикующий их в системе сопоставления LISP Mapping System. Обычно Map-Server не уполномочен отвечать на запросы, поэтому он пересылает их ETR. Однако сервер может работать в режиме посредника (proxy-mode), когда ETR делегирует ему право отвечать на запросы. Это полезно при ограниченности ресурсов ETR (например, CPU или питания).

#### **Map-Resolver - распознаватель сопоставлений**

Компонент сетевой инфраструктуры, соединяющий маршрутизатор ITR с системой сопоставления путём посредничества (proxying) для запросов, а иногда и откликов.

Интерфейс определяет 4 управляющих сообщения LISP, передаваемых в дейтаграммах UDP (порт 4342).

#### **Map-Register**

Эти сообщения применяются маршрутизаторами ETR для регистрации отображений в системе сопоставления и аутентифицируются с использованием общего для ETR и Map-Server ключа.

#### **Map-Notify**

По запросу ETR это сообщение передаёт Map-Server в ответ на сообщение Map-Register для подтверждения корректного получения отображения и передачи последнего состояния Map-Server для отображения EID на RLOC. В некоторых случаях Map-Notify может передаваться для прежних RLOC, если для EID зарегистрирован новый набор RLOC.

#### **Map-Request**

Это сообщение применяется маршрутизаторами ITR или распознавателями Map-Resolver для получения сопоставления данного EID.

#### **Map-Reply**

Это сообщение передаёт Map-Server или ETR в ответ на Map-Request, указывая найденное сопоставление. Следует отметить, что Map-Reply может содержать негативный отклик, если, например, запрошенный идентификатор EID не входит в пространство LISP EID. В таких случаях ITR обычно пересылает трафик как есть (без инкапсуляции) в сеть Internet. Это определено для поддержки поэтапного развёртывания LISP.

### 3.4.3. Система сопоставления

LISP архитектурно разделяет плоскости управления и данных с помощью стандартного интерфейса. Интерфейс связывает плоскость данных (маршрутизаторы, отвечающие за пересылку пакетов данных) с системой сопоставления LISP (база данных, отвечающая за хранение отображений). При таком разделении плоскости данных и управления могут иметь разную архитектуру и расширяться независимо. Обычно плоскость данных оптимизируется для маршрутизации пакетов по иерархическим адресам IP. Однако требования плоскости управления могут быть иными, например, за счёт использования преимуществ LCAF система сопоставления может хранить неиерархические ключи (такие как MAC-адреса) и для её расширяемости нужен будет другой подход. Другим важным различием плоскостей данных и управления LISP является то, что за счёт локального кэширования отображений в ITR системе сопоставления не требуется работать со скоростью линии.

При выборе архитектуры системы сопоставления было рассмотрено множество механизмов создания распределенных систем - базы данных на основе графов в форме альтернативной логической топологии LISP (LISP Alternative Logical Topology или LISP-ALT) [RFC6836], иерархические базы данных в форме дерева делегированных баз данных LISP (LISP Delegated Database Tree или LISP-DDT) [RFC8111], монолитные базы данных в форме LISP Not-so-novel EID-to-RLOC Database (LISP-NERD) [RFC6837], плоские базы данных в форме распределенной хэш-таблицы LISP (LISP Distributed Hash Table или LISP-DHT) [LISP-SHDHT] [Mathy], и основанные на групповой передаче базы данных [LISP-EMACS]. Следует отметить, что в некоторых вариантах, таких как приватное развёртывание, система сопоставления может быть логически централизованной и обычно будет включать один Map-Server/Map-Resolver.

В последующих параграфах более подробно рассмотрены две из упомянутых систем сопоставления (LISP-ALT и LISP-DDT), которые были реализованы и развёрнуты.

### 3.4.3.1. LISP-ALT

LISP-ALT [RFC6836] была первой системой сопоставления, которая была предложена, разработана и развёрнута в пилотной сети LISP. Она основана на распределенном наложении BGP с участием Map-Server и Map-Resolver. Узлы соединялись с партнёрами через статические туннели. Каждый вовлечённый Map-Server в топологии ALT анонсировал EID-Prefix, зарегистрированные обслуживаемыми ETR, что делало EID маршрутизируемыми в топологии ALT.

Когда маршрутизатору ITR нужно отображение, он передаёт Map-Request распознавателю Map-Resolver, который, используя топологию ALT, пересылает Map-Request в направлении Map-Server, отвечающего за сопоставление. При получении запроса Map-Server пересылает его маршрутизатору ETR, который отвечает напрямую ITR.

### 3.4.3.2. LISP-DDT

LISP-DDT [RFC8111] концептуально похожа на DNS - иерархический каталог, внутренняя структура которого отражает иерархическую природу адресного пространства EID. Иерархия DDT включает узлы DDT, формирующие дерево, листьями которого служат Map-Server. Наверху структуры размещается корень DDT, являющийся экземпляром узла DDT, соответствующим всему пространству адресов. Как и DNS, система DDT поддерживает множество избыточных узлов DDT и/или корней DDT. Распознаватели Map-Resolver являются клиентами иерархии DDT и могут обращаться к корню и другим узлам DDT.

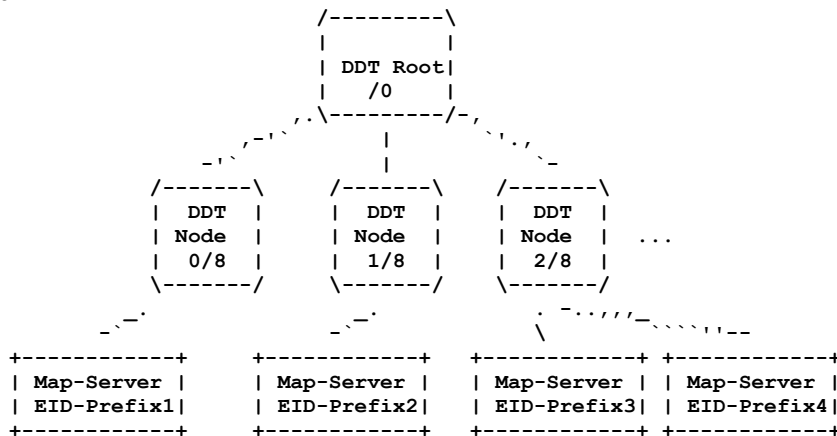


Рисунок 3. Схематическое представление дерева DDT.

Префиксы и структура на рисунке 3 представлены лишь для иллюстрации.

Структура DDT на деле не индексирует EID-Prefix, индексируя взамен расширенные префиксы (Extended EID-Prefix или XEID-Prefix), которые являются просто конкатенацией нескольких полей (от старшего бита к младшему): Database-ID, Instance ID, Address Family Identifier и фактических EID-Prefix. Database-ID указывается для возможных в будущем требований повышения уровня иерархии и возможности создания нескольких отдельных деревьев баз данных.

При ответах на запрос LISP-DDT работает подобно DNS, но поддерживает лишь интерактивный поиск. Клиенты DDT (обычно Map-Resolver) создают запросы Map-Request к корневому узлу DDT, получая в ответ новое управляющее сообщение LISP - Map-Referral. Это сообщение содержит список RLOC набора узлов DDT, соответствующих настроенному делегированию XEID. Т. е. сведения в Map-Referral указывают потомка запрашиваемого узла DDT, у которого есть более конкретные сведения об интересующем префиксе XEID-Prefix. Этот процесс повторяется, пока клиент DDT не пройдёт структуру дерева (вниз) и не найдет Map-Server, обслуживающий искомым XEID. В этот момент клиент передаёт Map-Request и получает Map-Reply с сопоставлениями. Важно подчеркнуть, что клиенты DDT могут кэшировать сведения из Map-Referral, т. е. структуру DDT, чтобы сократить время извлечения сопоставлений [Jakab].

Система сопоставления DDT основана на ручной настройке, т. е. на Map-Resolver настраивается набор доступных корневых узлов DDT, а на узлах DDT - соответствующее делегирование XEID. Изменение настроек на узлах DDT требуется лишь при смене структуры дерева и настройки не зависят от динамики EID (выделения RLOC или смены правил TE).

## 3.5. Механизмы межсетевого взаимодействия

EID обычно идентичны адресам IPv4 или IPv6 и хранятся в системе сопоставления LISP. Однако они обычно не анонсируются в систему маршрутизации за пределами локального домена LISP. Поэтому для LISP нужен механизм межсетевого взаимодействия, позволяющий сайтам LISP взаимодействовать с сайтами, не понимающими LISP (и наоборот). Механизмы межсетевого взаимодействия LISP заданы в [RFC6832].

В LISP заданы две сущности для обеспечения межсетевого взаимодействия.

### Proxy Ingress Tunnel Router (PITR) - маршрутизатор-посредник входного туннеля

PITR обеспечивают связность унаследованной сети Internet с сайтами LISP. PITR анонсируют в систему глобальной маршрутизации блоки EID-Prefix (с агрегированием при возможности) для привлечения трафика. Для

каждого входящего пакета из источника вне сайта LISP (не EID) PITR использует инкапсуляцию LISP в направлении RLOC подходящего сайта LISP. Влияние PITR на размер таблицы маршрутизации зоны без принятого по умолчанию маршрута (Default-Free Zone или DFZ) в худшем случае похоже на ситуацию без применения LISP. EID-Prefix будут по возможности агрегироваться как PITR, так и системой глобальной маршрутизации.

#### **Proxy Egress Tunnel Router (PETR) - маршрутизатор-посредник выходного туннеля**

PETR обеспечивает связность сайта LISP с традиционной сетью Internet. В некоторых случаях сайты LISP не могут передавать инкапсулированные пакеты с локальным адресом EID в качестве источника в традиционную сеть Internet, например, когда краевой маршрутизатор провайдера (Provider Edge или PE) использует индивидуальную пересылку по обратному пути (Unicast Reverse Path Forwarding или uRPF) или промежуточная сеть между сайтом LISP и не понимающим LISP сайтом не поддерживает нужную версию IP (IPv4 или IPv6). В обоих PETR преодолевает такие ограничения за счёт инкапсуляции пакетов. Не существует конкретных правил распространения адресов PETR RLOC маршрутизаторам ITR.

Кроме того, LISP определяет механизмы для работы с приватными EID [RFC1918] через LISP-NAT [RFC6832]. В этом случае xTR заменяет приватное значение EID источника на маршрутизируемое. На момент написания документа продолжалась работа по прохождению через NAT, т. е. размещения xTRs за NAT с использованием немаршрутизируемых RLOC.

PITR, PETR, и LISP-NAT позволяют развёртывать LISP поэтапно, обеспечивая достаточную гибкость размещения временных между частями сети с поддержкой и без поддержки LISP и упрощая перенос этих границ со временем.

## **4. Механизмы работы LISP**

В этом разделе описаны основные рабочие механизмы, определённые в LISP.

### **4.1. Поддержка кэша**

Разделение плоскостей управления и данных в LISP с хранением сопоставлений в плоскости данных и использовании их для пересылки в плоскости управления требует локального кэширования в ITR для снижения сигнальных издержек (Map-Request/Map-Reply) и повышения скорости пересылки. Локальный кэш в ITR называют Map-Cache и применяют для пакетов с инкапсуляцией LISP. Map-Cache индексируется по парам (Instance ID, EID-Prefix) и содержит в основном набор RLOC со связанными правилами TE (приоритет и вес).

Для Map-Cache, как и иных случаев кэширования, нужны механизмы обеспечения когерентности, чтобы сведения оставались актуальными. LISP задает 3 основных механизма поддержки когерентности кэша.

#### **Record Time To Live (TTL) - срок действия записи**

Каждая запись сопоставления имеет поле TTL, задаваемое маршрутизатором ETR. По истечении TTL запись не может применяться ITR, пока не будет обновлена отправкой нового Map-Request.

#### **Solicit-Map-Request (SMR) - запрос Map-Request**

SMR служит явным механизмом обновления данных сопоставления. В частности, можно передать специальный тип Map-Request по требованию ETR для запроса обновления сопоставлений. При получении сообщения SMR маршрутизатор ITR должен обновить привязки передавая Map-Request системе сопоставления. Использование SMR описано в [RFC9301].

#### **Map-Versioning - версия сопоставления**

Этот необязательный механизм добавляет в конец заголовка LISP пакетов данных номер версии отображения, используемой маршрутизатором xTR. При получении xTR пакета с инкапсуляцией LISP от удалённого xTR, можно узнать, не устарела ли локальная или удалённая версия Map-Cache. Если локальная версия устарела маршрутизатор передаёт Map-Request для удалённого EID, чтобы получить новое сопоставление. Если же обнаружено устаревание Map-Cache на удалённом xTR, маршрутизатор передаёт SMR для уведомления того о доступности нового сопоставления. Подробное описание процесса приведено в [RFC9302].

В некоторых случаях запись Map-Cache можно обновить заранее с помощью описанных ниже механизмов.

### **4.2. Достижимость RLOC**

В большинстве случаев LISP работает с системой отображения на основе вытягивания (pull), например, DDT. Это обеспечивает сквозную архитектуру вытягивания и состояние сети храниться в плоскости управления, а плоскость данных извлекает его по запросу. Это влияет на распространение сведений о достижимости и живучести xTR, поскольку архитектура вытягивания требует явных механизмов распространения таких данных. Поэтому в LISP задан набор механизмов для информирования ITR и PITR о доступности кэшированных RLOC.

#### **Locator-Status-Bits (LSBs) - биты состояния локатора**

Метод LSBs является пассивным. Поле LSB передаётся в заголовках LISP пакетов данных и может устанавливаться маршрутизаторами ETR для указания активных и неактивных (up/down) RLOC сайта ETR. Эти сведения могут служить маршрутизаторам ITR подсказкой о доступности для выполнения дополнительной проверки. LSB не указывают статус доступности пути, а лишь дают подсказки о состоянии RLOC. Поэтому они не должны применяться в общедоступной сети Internet и следует связывать их с Map-Versioning для предотвращения «гонки», где LSB считаются относящимися не к тем RLOC, с которыми они связаны.

#### **Echo-Nonce**

Это пассивный метод, который может эффективно работать лишь при двухсторонних потоках данных между двумя взаимодействующими xTR. По сути, ITR добавляет в конце заголовка LISP пакетов данных случайное значение (nonce). Если путь и зондируемый локатор активны, ETR будет добавлять это же случайное значение в следующий пакет данных. Если этого не происходит, ITR может считать локатор недоступным. При одностороннем трафике или несовпадении принимающего трафик ETR с передающим трафик назад маршрутизатором ITR нужны дополнительные механизмы. Механизм Echo-Nonce должен применяться лишь в доверенных средах, а не в общедоступной сети Internet.

#### **RLOC-Probing - зондирование RLOC**

Имеется механизм активного зондирования, когда ITR передают пробные пакеты конкретным локаторам. Это позволяет проверить как локатор, так и путь к нему. В частности, это делается путём отправки Map-Request (с некоторыми флагами) в плоскости данных (пространство RLOC) и ожидания Map-Reply, передаваемого также в плоскости данных. Активная природа RLOC-Probing обеспечивает эффективный механизм определения

доступности и (в случае отказа) переключения на другой локатор. Кроме того, механизм обеспечивает полезную оценку RTT для задержек на пути, которую могут использовать другие сетевые алгоритмы.

Следует отметить, что RLOC-Probing и Echo-Nonce могут работать вместе. В частности, если nonce не возвращается, ITR не может определить на каком из направлений возник отказ. В этом случае ITR может использовать RLOC-Probing.

LISP также рекомендует делать вывод о доступности локаторов из сведений, предоставляемых базовой сетью.

#### **Сигналы ICMP**

Базовая среда LISP - инфраструктура Internet - использует ICMP для сигналов о недоступности (среди прочего). LISP может использовать это и считать получение сообщений ICMP Network Unreachable или ICMP Host Unreachable подсказкой о возможной недоступности локатора, вызывающей дополнительную проверку.

#### **Базовая маршрутизация**

Протоколы BGP и IGP несут сведения о доступности. Маршрутизаторы с поддержкой LISP, имеющие доступ к данным базовой маршрутизации, могут использовать их для проверки доступности локатора.

### **4.3. Синхронизация ETR**

Все ETR, полномочные для конкретного EID-Prefix, должны анонсировать запрашивающим одно и то же сопоставление. Это значит, что ETR должны знать о статусе RLOC в других ETR. Это называют синхронизацией ETR.

На момент написания документа в LISP не был задан механизм синхронизации ETR. Хотя для решения задачи пригодны многие имеющиеся методы, вопрос ещё не решён. Поэтому операторы должны полагаться на согласованную ручную настройку.

### **4.4. Обработка MTU**

Поскольку LISP инкапсулирует пакеты, протоколу приходится иметь дело с превышением размера пакетов над MTU для пути между ITR и ETR. LISP определяет два механизма.

#### **Без учёта состояния**

Действующим считается значение MTU с точки зрения ITR. Если пакет слишком велик для этого значения и может быть фрагментирован, содержимое фрагментируется на ITR, а сборка выполняется хостом-получателем.

#### **С учётом состояния**

ITR отслеживают значения MTU на путях к целевым локаторам на основании пакетов ICMP Too Big от промежуточных маршрутизаторов. Маршрутизаторы ITR передают сообщения ICMP Too Big для информирования отправителей об эффективном значении MTU. Кроме того, ITR могут применять механизмы определения MTU для пути (Path MTU Discovery или PMTUD) [RFC1191] или определение MTU на уровне пакетизации (Packetization Layer Path MTU Discovery или PLPMTUD) [RFC4821] для отслеживания MTU в направлении локаторов.

В обоих случаях при невозможности фрагментировать пакеты (IPv4 с DF=1 или IPv6) ITR отбрасывает пакет и передаёт его отправителю сообщение ICMP Too Big.

## **5. Мобильность**

Разделение локаторов и идентификаторов в LISP подходит для целей TE, где сайты LISP могут менять свои точки присоединения к Internet (т. е. RLOC) не затрачивая конечные точки и ядро Internet. В этом контексте граничные маршрутизаторы используют функциональность xTR, а конечные точки не знают о наличии LISP. Это похоже на Network Mobility [RFC3963], однако данный режим работы не обеспечивает «бесшовной» мобильности конечных точек между разными сайтами LISP, поскольку адрес EID может не маршрутизироваться в посещенном сайте. Тем не менее, LISP можно использовать для бесшовной мобильности IP, когда LISP реализован непосредственно на конечной точке или та перемещается к подключённому xTR. Тогда каждая конечная точка является xTR, а адрес EID - это один из предоставленных сетевому стеку, используемому приложениями, тогда как RLOC берётся из посещенной сети. Это похоже на функциональность Mobile IP ([RFC5944] и [RFC6275]).

При смене устройством своего локатора RLOC маршрутизатор xTR обновляет RLOC в своём локальном сопоставлении и регистрирует его на своём Map-Server, обычно с малым значением TTL (1 минута). Чтобы избежать потребности в домашнем шлюзе, ITR также указывает смену RLOC всем удаленным устройствам, имеющим текущую связь с перемещенным устройством. Сочетание обоих методов обеспечивает расширяемость системы, поскольку сигнализация строго ограничена Map-Server и хостами, с которыми имеется связь. В случае перемещения EID-Prefix может быть достаточно мелким, вплоть до /32 или /128 (IPv4 и IPv6, соответственно), в зависимости от конкретного применения (например, перенос подсети или одной виртуальной машины или мобильного узла).

Разделение идентификатора и локатора, обеспечиваемое LISP, позволяет работать с другими решениями для мобильности L2 и L3.

## **6. Групповая передача**

LISP поддерживает доставку групповых пакетов IP, переданных из пространства EID. Требуемые изменения протоколов групповой доставки указаны в [RFC6831].

LISP может создавать состояния групповой рассылки (для приёма и передачи) как в ядре, так и на сайтах. При использовании сигнализации для создания групповой рассылки на сайтах маршрутизаторы LISP инкапсулируют сообщения PIM Join/Prune от получателя к сайтам источников как индивидуальные пакеты. В ядре маршрутизаторы ETR создают новое сообщение PIM Join/Prune, адресованное RLOC маршрутизатора ITR, обслуживающего источник. Упрощённая последовательность приведена ниже.

1. Конечный хост, желающий присоединиться к групповому каналу передаёт отчёт IGMP. Групповые маршрутизаторы PIM на сайте LISP распространяют сообщения PIM Join/Prune (S-EID, G) в направлении ETR.
2. Сообщение Join поступает в ETR и маршрутизатор ETR создаёт два сообщения Join. Первое является индивидуальным пакетом с инкапсуляцией LISP для исходного сообщения Join в направлении RLOC обслуживающего источник маршрутизатора ITR. Это сообщение создаёт групповое состояние (S-EID, G) на сайте источника. Второе сообщение Join содержит в качестве адреса получателя RLOC маршрутизатора ITR, обслуживающего источник (S-RLOC, G) и создаёт состояние групповой рассылки в ядре.



3. Групповые пакеты данных, созданные источником (S-EID, G) передаются от него к ITR. Маршрутизатор ITR инкапсулирует групповые пакеты в LISP. Внешний заголовок включает его RLOC (S-RLOC) как отправителя и исходный адрес multicast-группы (G) как получателя. Отметим, что групповые адреса являются логическими и не распознаются системой сопоставления. Затем групповые пакеты передаются через ядро в направлении принимающих ETR, которые деинкапсулируют пакеты и пересылают их по групповому состоянию на сайте.

Отметим, что внутренние и внешние групповые адреса обычно различаются за исключением особых случаев, когда базовый провайдер жёстко контролирует наложение. Спецификации LISP уже поддерживают все режимы PIM modes [RFC6831]. Кроме того, LISP может поддерживать другие механизмы сохранения групповых состояний.

Когда групповые источники и получатели имеются на сайтах LISP, а ядро сети между сайтами не обеспечивает поддержки групповой передачи, можно применять бессигнальный механизм для создания наложения, которое позволит передавать групповой трафик между сайтами и соединить деревья групповой рассылки на разных сайтах [RFC8378]. Регистрации с разных сайтов-получателей будут слиты в системе сопоставления для сборки списка групповой репликации, включающего все RLOC, ведущие к получателям для конкретной группы или группового канала. Список репликации для каждой конкретной групповой записи поддерживается как запись системы сопоставления LISP.

## 7. Варианты применения

### 7.1. Организация трафика

Сайт LISP может жёстко указывать ETR, через которые трафик должен входить в сеть сайта, даже когда путь к ETR не контролируется сайтом LISP. Этот точный контроль реализуется в сопоставлениях. Когда удалённый сайт хочет передать трафик сайту LISP, он просматривает отображения, связанные с целевым EID, в системе сопоставления. Отображение передаётся напрямую уполномоченным ETR для EID и не меняется в промежуточных сетях.

Отображение связывает список RLOC с EID-Prefix. Каждый локатор RLOC соответствует интерфейсу ETR (или набора ETR), способному корректно переслать трафик по EID из префикса. Для каждого RLOC в отображении указывается приоритет и вес. Приоритет служит для выбора предпочтительных для передачи пакетов RLOC (менее предпочтительные предоставляются для резервирования), а вес позволяет распределять нагрузку между RLOC с одинаковым приоритетом пропорционально заданному значению веса.

Поскольку сопоставления выдаются непосредственно полномочными ETR для EID и не меняются при передаче на удалённый сайт, это обеспечивает высокий уровень гибкости TE для входящего междоменного трафика и позволяет сайту задавать свои правила сопоставления для каждого удалённого сайта.

### 7.2. LISP для сосуществования с IPv6

Инкапсуляция LISP позволяет доставлять пакеты, использующие EID из данного пространства адресов (например, IPv6) в пакетах другого семейства адресов (например, IPv4). Отсутствие привязки между семействами адресов RLOC и EID делает LISP подходящим кандидатом, позволяющим, например, развернуть IPv6 через ядро без поддержки IPv6.

Например, два ЦОД, поддерживающие лишь IPv6 можно соединить через традиционную сеть IPv4 (Internet). Если граничные маршрутизаторы поддерживают LISP, передача пакетов между ЦОД выполняется без какой-либо трансляции, поскольку исходные пакеты IPv6 (в пространстве EID) LISP будет инкапсулировать и передавать в пакетах IPv4 традиционной сети Internet через IPv4 RLOC.

### 7.3. LISP для VPN

Обычно в одной физической инфраструктуре работает несколько виртуальных сетей. В таких ситуациях определение принадлежности пакета к виртуальной сети становится важной задачей и для этого часто применяются теги или метки. При использовании LISP пакеты можно различать по полю Instance ID. Когда ITR инкапсулирует пакет из определённой виртуальной сети (например, известной по VRF<sup>1</sup> или VLAN), он помечает инкапсулированный пакет значением Instance ID, соответствующим виртуальной сети. Когда ETR получает пакет с Instance ID, он использует это поле для определения принадлежности пакета.

Основные применения LISP для виртуальных частных сетей не вносят дополнительных требований к базовой сети, коль скоро она работает по протоколу IP.

### 7.4. LISP для мобильности VM в ЦОД

Способ обеспечения бесшовной мобильности виртуальных машин (VM) в ЦОД состоит в представлении опорной сети ЦОД как пространства RLOC, а подсетей с серверами, как пространство EID. Маршрутизаторы LISP помещаются на границе между магистралью и каждой подсетью серверов. При перемещении VM в другую подсеть она может (временно) сохранить прежний адрес для продолжения работы без сброса транспортных соединений. Когда xTR обнаруживает адрес источника из другой подсети, он регистрирует адрес в системе сопоставления. Для информирования других маршрутизаторов LISP о переносе виртуальной машины и её новом местоположении, чтобы избежать обходных путей через исходную подсеть, применяются такие механизмы, как сообщения Solicit-Map-Request.

## 8. Вопросы безопасности

В системах сопоставления с выталкиванием (push) требуемые для пересылки состояния изучаются непосредственно по трафику. Однако в архитектуре с вытягиванием (pull) система становится реактивной и события в плоскости данных (например, прибытие пакета с неизвестным адресом получателя) могут вызывать события в плоскости управления. Такое обычение по потребности обеспечивает много преимуществ, но может влиять на безопасность.

Обычно плоскость данных реализуется на быстром пути маршрутизаторов для обеспечения высокоскоростной пересылки, тогда как плоскость управления реализуется на медленном пути маршрутизаторов для обеспечения гибкости. Разрыв производительности медленного и быстрого пути может составлять несколько порядков. Поэтому

<sup>1</sup>Virtual Routing and Forwarding - виртуальная маршрутизация и пересылка.

нужно тщательно обдумать способ уведомления плоскости управления о событиях в плоскости данных, чтобы не перегрузить медленный путь, следует также применять ограничение скорости, как указано в [RFC9300] и [RFC9301].

Следует соблюдать осторожность, чтобы не перегрузить систему сопоставления (т. е. инфраструктуру плоскости управления), поскольку выполняемые ею операции могут быть сложнее операций плоскости данных. По этой причине [RFC9300] и [RFC9301] рекомендуют ограничивать скорость передачи сообщений в систему сопоставления.

Для повышения отказоустойчивости и сокращения обмена сообщениями LISP позволяет передачу некоторых сведений, таких как доступность локаторов, напрямую в пакетах плоскости данных. В средах без полного доверия, подобных сети Internet, данные управления из пакетов плоскости данных использовать недопустимо или их следует проверять перед употреблением.

Сопоставления являются центральным элементом LISP и должны приниматься все меры предосторожности для предотвращения вредоносных манипуляций или использования их злоумышленниками. Использование доверенных серверов Map-Server, строго соответствующих [RFC9301], и механизмов аутентификации, предоставляемых LISP-SEC [RFC9303], снижает риск атак на целостность сопоставлений. В более критических средах могут потребоваться меры защиты. Реализация защиты для данной системы сопоставления сильно зависит от архитектуры этой системы и модели угроз, предполагаемой для развёртывания. Поэтому безопасность системы сопоставления должна рассматриваться в соответствующих документах по архитектуре Mapping System.

Как и в других механизмах туннелирования промежуточные устройства на пути между ITR (или PITR) и ETR (или PETR) должны иметь механизмы снятия инкапсуляции для корректной проверки содержимого пакетов с инкапсуляцией LISP.

Как и другие механизмы инкапсуляции и сопоставления (map-and-encap), LISP разрешает «трехстороннюю» маршрутизацию (т. е. поток пакетов проходит через разные граничные маршрутизаторы в зависимости от направления). Это означает, что у промежуточных устройств может не быть полного представления о трафике, который они инспектируют или обрабатывают. Кроме того, пакеты с инкапсуляцией LISP маршрутизируются по внешним адресам IP (RLOC) и могут быть доставлены маршрутизатору ETR, который не отвечает за EID получателя пакета, или сетевому элементу, не являющемуся ETR. Снижение риска обеспечивается применением подходящих методов фильтрации, на элементах сети, которые могут получать неожиданные пакеты с инкапсуляцией LISP.

Дополнительные сведения о влиянии LISP на безопасность приведены в [RFC7835].

## 9. Взаимодействие с IANA

Этот документ не требует действий IANA.

## 10. Литература

### 10.1. Нормативные документы

- [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", [RFC 1191](#), DOI 10.17487/RFC1191, November 1990, <<https://www.rfc-editor.org/info/rfc1191>>.
- [RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and E. Lear, "Address Allocation for Private Internets", BCP 5, [RFC 1918](#), DOI 10.17487/RFC1918, February 1996, <<https://www.rfc-editor.org/info/rfc1918>>.
- [RFC2992] Hopps, C., "Analysis of an Equal-Cost Multi-Path Algorithm", [RFC 2992](#), DOI 10.17487/RFC2992, November 2000, <<https://www.rfc-editor.org/info/rfc2992>>.
- [RFC3963] Devarapalli, V., Wakikawa, R., Petrescu, A., and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol", RFC 3963, DOI 10.17487/RFC3963, January 2005, <<https://www.rfc-editor.org/info/rfc3963>>.
- [RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", [RFC 4821](#), DOI 10.17487/RFC4821, March 2007, <<https://www.rfc-editor.org/info/rfc4821>>.
- [RFC4984] Meyer, D., Ed., Zhang, L., Ed., and K. Fall, Ed., "Report from the IAB Workshop on Routing and Addressing", RFC 4984, DOI 10.17487/RFC4984, September 2007, <<https://www.rfc-editor.org/info/rfc4984>>.
- [RFC5944] Perkins, C., Ed., "IP Mobility Support for IPv4, Revised", [RFC 5944](#), DOI 10.17487/RFC5944, November 2010, <<https://www.rfc-editor.org/info/rfc5944>>.
- [RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, DOI 10.17487/RFC6275, July 2011, <<https://www.rfc-editor.org/info/rfc6275>>.
- [RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, DOI 10.17487/RFC6831, January 2013, <<https://www.rfc-editor.org/info/rfc6831>>.
- [RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, DOI 10.17487/RFC6832, January 2013, <<https://www.rfc-editor.org/info/rfc6832>>.
- [RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, DOI 10.17487/RFC6835, January 2013, <<https://www.rfc-editor.org/info/rfc6835>>.
- [RFC6836] Fuller, V., Farinacci, D., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, DOI 10.17487/RFC6836, January 2013, <<https://www.rfc-editor.org/info/rfc6836>>.
- [RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, DOI 10.17487/RFC6837, January 2013, <<https://www.rfc-editor.org/info/rfc6837>>.
- [RFC6935] Eubanks, M., Chimento, P., and M. Westerlund, "IPv6 and UDP Checksums for Tunneled Packets", RFC 6935, DOI 10.17487/RFC6935, April 2013, <<https://www.rfc-editor.org/info/rfc6935>>.
- [RFC6936] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", RFC 6936, DOI 10.17487/RFC6936, April 2013, <<https://www.rfc-editor.org/info/rfc6936>>.

- [RFC7052] Schudel, G., Jain, A., and V. Moreno, "Locator/ID Separation Protocol (LISP) MIB", RFC 7052, DOI 10.17487/RFC7052, October 2013, <<https://www.rfc-editor.org/info/rfc7052>>.
- [RFC7215] Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "Locator/Identifier Separation Protocol (LISP) Network Element Deployment Considerations", RFC 7215, DOI 10.17487/RFC7215, April 2014, <<https://www.rfc-editor.org/info/rfc7215>>.
- [RFC7835] Saucez, D., Iannone, L., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Threat Analysis", RFC 7835, DOI 10.17487/RFC7835, April 2016, <<https://www.rfc-editor.org/info/rfc7835>>.
- [RFC8060] Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", RFC 8060, DOI 10.17487/RFC8060, February 2017, <<https://www.rfc-editor.org/info/rfc8060>>.
- [RFC8111] Fuller, V., Lewis, D., Ermagan, V., Jain, A., and A. Smirnov, "Locator/ID Separation Protocol Delegated Database Tree (LISP-DDT)", RFC 8111, DOI 10.17487/RFC8111, May 2017, <<https://www.rfc-editor.org/info/rfc8111>>.
- [RFC8378] Moreno, V. and D. Farinacci, "Signal-Free Locator/ID Separation Protocol (LISP) Multicast", RFC 8378, DOI 10.17487/RFC8378, May 2018, <<https://www.rfc-editor.org/info/rfc8378>>.
- [RFC9300] Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos, Ed., "The Locator/ID Separation Protocol (LISP)", RFC 9300, DOI 10.17487/RFC9300, October 2022, <<https://www.rfc-editor.org/info/rfc9300>>.
- [RFC9301] Farinacci, D., Maino, F., Fuller, V., and A. Cabellos, Ed., "Locator/ID Separation Protocol (LISP) Control Plane", RFC 9301, DOI 10.17487/RFC9301, October 2022, <<https://www.rfc-editor.org/info/rfc9301>>.
- [RFC9302] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 9302, DOI 10.17487/RFC9302, October 2022, <<https://www.rfc-editor.org/info/rfc9302>>.
- [RFC9303] Maino, F., Ermagan, V., Cabellos, A., and D. Saucez, "Locator/ID Separation Protocol Security (LISP-SEC)", RFC 9303, DOI 10.17487/RFC9303, October 2022, <<https://www.rfc-editor.org/info/rfc9303>>.

## 10.2. Дополнительная литература

- [Jakab] Jakab, L., Cabellos-Aparicio, A., Coras, F., Saucez, D., and O. Bonaventure, "LISP-TREE: A DNS Hierarchy to Support the LISP Mapping System", IEEE Journal on Selected Areas in Communications, vol. 28, no. 8, pp. 1332-1343, DOI 10.1109/JSAC.2010.101011, October 2010, <<https://ieeexplore.ieee.org/document/5586446>>.
- [LISP-EMACS] Brim, S., Farinacci, D., Meyer, D., and J. Curran, "EID Mappings Multicast Across Cooperating Systems for LISP", Work in Progress, Internet-Draft, draft-curran-lisp-emacs-00, 9 November 2007, <<https://www.ietf.org/archive/id/draft-curran-lisp-emacs-00.txt>>.
- [LISP-SHDHT] Cheng, L. and M. Sun, "LISP Single-Hop DHT Mapping Overlay", Work in Progress, Internet-Draft, draft-cheng-lisp-shdht-04, 15 July 2013, <<https://www.ietf.org/archive/id/draft-cheng-lisp-shdht-04.txt>>.
- [Mathy] Mathy, L. and L. Iannone, "LISP-DHT: Towards a DHT to map identifiers onto locators", CoNEXT '08: Proceedings of the 2008 ACM CoNEXT Conference, ReArch '08 — Re-Architecting the Internet, DOI 10.1145/1544012.1544073, December 2008, <<https://dl.acm.org/doi/10.1145/1544012.1544073>>.
- [Quoitin] Quoitin, B., Iannone, L., de Launois, C., and O. Bonaventure, "Evaluating the Benefits of the Locator/Identifier Separation", Proceedings of 2nd ACM/IEEE International Workshop on Mobility in the Evolving Internet Architecture, DOI 10.1145/1366919.1366926, August 2007, <<https://dl.acm.org/doi/10.1145/1366919.1366926>>.

## Приложение А. История разделения идентификаторов и локаторов

Архитектура LISP для разделения местоположения и отождествления стала результатом дискуссий во время семинара IAB Routing and Addressing в Амстердаме в октябре 2006 г. [RFC4984].

Сразу после семинара спонтанно сформировалась небольшая группа единомышленников для работы над идеей, возникшей в неформальных дискуссиях на семинаре в различных почтовых конференциях. Первый документ Internet-Draft для LISP появился в январе 2007 г.

В это время начались пробные реализации, а первые пробные развёртывания состоялись в июне 2007 г. Результаты этих экспериментов учитывались при проектировании, занявшем несколько лет. На данный момент LISP представляет собой умеренно зрелую систему, прошедшую длинную цепочку изменений и обновлений.

Протокол LISP перешёл из сферы IRTF в рабочую группу IETF в марте 2009 г. После многочисленных исправления базовые спецификации в начале 2013 г. были выпущены как RFC. Работа по расширению, совершенствованию и поиску новых применений продолжается (и несомненно будет длиться ещё долго). Рабочая группа LISP была в 2018 г. преобразована для продолжения работы над базовым протоколом LISP и создания документов Standards Track.

### А.1. Старые модели LISP

LISP по начальной задумке имел несколько возможных режимов работы, называемых моделями. Хотя они больше не применяются, иногда о них вспоминают, поэтому модели перечислены здесь.

#### LISP 1

Все EID появляются в обычных таблицах маршрутизации и пересылки в сети (т. е. являются «маршрутизируемыми»). Это свойство применяется для получения сопоставлений EID с RLOC в процессе начальной загрузки (bootstrapping). Пакеты передаются с EID в качестве адресата во внешней заголовке и когда ETR видит такой пакет, он передаёт Map-Reply маршрутизатору ITR источника, давая полное сопоставление.

#### LISP 1.5

LISP 1.5 похож на LISP 1, но маршрутизация EID происходит в отдельной сети.

#### LISP 2

EID не маршрутизируются, сопоставления EID с RLOC доступны через DNS.

**LISP 3**

EID не маршрутизируются и их нужно искать в новой базе данных о сопоставлениях EID с RLOC (изначально это была система, использующая распределенные хэш-таблицы). Были возможны два варианта - push-системы, где все отображения распространялись всем ITR, и pull-системы, в которых ITR загружали сопоставления при необходимости.

**Благодарности**

Этот документ инициировал Noel Chiappa и большая часть идей представлена им. Авторы признательны за его важный вклад в работу и благодарны ему.

Спасибо также Dino Farinacci, Fabio Maino, Luigi Iannone, Sharon Barkai, Isidoros Kouvelas, Christian Cassar, Florin Coras, Marc Binderberger, Alberto Rodriguez-Natal, Ronald Bonica, Chad Hintz, Robert Raszuk, Joel M. Halpern, Darrel Lewis, David Black.

**Адреса авторов****Albert Cabellos**

Universitat Politecnica de Catalunya  
c/ Jordi Girona s/n  
08034 Barcelona  
Spain  
Email: [acabello@ac.upc.edu](mailto:acabello@ac.upc.edu)

**Damien Saucez** (editor)

Inria  
2004 route des Lucioles - BP 93  
Sophia Antipolis  
France  
Email: [damien.saucez@inria.fr](mailto:damien.saucez@inria.fr)

**Перевод на русский язык**

Николай Малых

[nmalykh@protokols.ru](mailto:nmalykh@protokols.ru)