



IP Routing: LISP Configuration Guide, Cisco IOS XE Everest 16.5

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com go trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2019 Cisco Systems, Inc. All rights reserved.



CONTENTS

CHAPTER 1

Read Me First 1

CHAPTER 2

Locator ID Separation Protocol (LISP) Overview 3

- Finding Feature Information 3
- Prerequisites for Configuring LISP 3
- Restrictions for Configuring LISP 4
- Information About Configuring LISP 4
 - LISP Functionality Overview 4
 - LISP Network Element Functions 5
 - LISP Alternative Logical Topology 5
 - LISP Egress Tunnel Router 5
 - LISP Ingress Tunnel Router (ITR) 6
 - LISP Map Resolver 6
 - LISP Map Server 7
 - LISP Proxy ETR 7
 - LISP Proxy ITR 7
 - Feature Information for LISP Overview 8

CHAPTER 3

Configuring LISP (Locator ID Separation Protocol) 11

- Prerequisites for Configuring LISP 11
- How to Configure LISP 11
 - Configure a Dual-Homed LISP Site with Two IPv4 RLOCs and an IPv4 EID 11
 - Configure a Multihomed LISP Site with Two xTRs and Two IPv4 RLOCs and an IPv4 EID 16
 - Configure a Multihomed LISP Site with Two xTRs and Two IPv4 RLOCs and Both an IPv4 and an IPv6 EID 23

Configure a Multihomed LISP Site with Two xTRs that Each have Both an IPv4 and an IPv6 RLOC and Both an IPv4 and an IPv6 EID	32
Configure a Private LISP Mapping System Using a Standalone Map Resolver/Map Server	42
Configure a Public Mapping System Using Separate ALT-Connected Map Resolver and Map Server Devices	48
Configuring an ALT-Connected LISP Map Resolver	48
Configuring an ALT-Connected LISP Map Server	55
Configure a PETR and a PITR	65
Deploying a Proxy Egress Tunnel Router with both an IPv4 and an IPv6 RLOC	65
Deploying a Proxy Ingress Tunnel Router with both an IPv4 and an IPv6 RLOC	68
Verify and Troubleshoot Locator ID Separation Protocol	77
Additional References for Configuring LISP	84
Feature Information for LISP	85

CHAPTER 4**LISP Multicast 87**

Finding Feature Information	87
Prerequisites for LISP Multicast	87
Restrictions for LISP Multicast	88
Information About LISP Multicast	88
How to Configure LISP Multicast	89
Configuring LISP Multicast	89
Configuring LISP Multicast in VRFs	91
Verifying LISP Multicast	93
Configuration Examples for LISP Multicast	95
Example: Configuring LISP Multicast	95
Example: Configuring LISP Multicast in VRFs	101
Additional References for LISP Multicast	101
Feature Information for LISP Multicast	103

CHAPTER 5**LISP Shared Model Virtualization 105**

Finding Feature Information	105
Information About LISP Shared Model Virtualization	105
Overview of LISP Virtualization	105
LISP Shared Model Virtualization	108

LISP Shared Model Virtualization Architecture	109
LISP Shared Model Virtualization Implementation Considerations and Caveats	110
How to Configure LISP Shared Model Virtualization	110
Configure Simple LISP Shared Model Virtualization	110
Configuring a Private LISP Mapping System for LISP Shared Model Virtualization	117
Configure Large-Scale LISP Shared Model Virtualization	120
Configure a Remote Site for Large-Scale LISP Shared Model Virtualization	130
Verifying and Troubleshooting LISP Virtualization	136
Configuration Examples for LISP Shared Model Virtualization	142
Additional References	142
Feature Information for LISP Shared Model Virtualization	144

CHAPTER 6**LISP Parallel Model Virtualization 145**

Finding Feature Information	145
Information About LISP Parallel Model Virtualization	145
Overview of LISP Virtualization	145
LISP Parallel Model Virtualization	148
LISP Parallel Model Virtualization Architecture	149
LISP Parallel Model Virtualization Implementation Considerations and Caveats	149
How to Configure LISP Parallel Model Virtualization	150
Configure Simple LISP Parallel Model Virtualization	150
Configuring a Private LISP Mapping System for LISP Parallel Model Virtualization	157
Verifying and Troubleshooting LISP Virtualization	162
Configuration Examples for LISP Parallel Model Virtualization	168
Additional References	168
Feature Information for LISP Parallel Model Virtualization	169

CHAPTER 7**LISP Host Mobility Across Subnet 171**

Finding Feature Information	171
Information About LISP Host Mobility Across Subnet	171
Overview of LISP Host Mobility Across Subnet	171

CHAPTER 8**LISP Delegate Database Tree (DDT) 173**

Finding Feature Information	173
-----------------------------	-----

Information About Delegate Database Tree (DDT)	173
Overview of LISP Delegate Database Tree (DDT)	173

CHAPTER 9

LISP ESM Multihop Mobility	175
Finding Feature Information	175
Restrictions for LISP ESM Multihop Mobility	175
Information About LISP ESM Multihop Mobility	176
LISP ESM Multihop Mobility Overview	176
How to Configure LISP ESM Multihop Mobility	178
Configuring First-Hop Router	178
Configuring Site Gateway xTR	181
Configuring xTR	184
Configuring Map Server Map Resolver	186
Configuration Examples for LISP ESM Multihop Mobility	188
Example: First-Hop Router Configuration	188
Example: Site Gateway xTR Configuration	189
Example: xTR Configuration	189
Example: Map Server Map Resolver Configuration	189
Additional References for LISP ESM Multihop Mobility	190
Feature Information for LISP ESM Multihop Mobility	190

CHAPTER 10

LISP Support for Disjoint RLOC Domains	191
Finding Feature Information	191
Prerequisites for LISP Support for Disjoint RLOC Domains	191
Restrictions for LISP Support for Disjoint RLOC Domains	192
Information About LISP Support for Disjoint RLOC Domains	192
LISP Support for Disjoint RLOC Domains Overview	192
How to configure LISP Support for Disjoint RLOC Domains	194
Configuring xTR	194
Configuring MSMR	197
Configuring RTR	201
Verifying LISP Support for Disjoint RLOC Domains	205
Configuration Examples for LISP Support for Disjoint RLOC Domains	206
Example: Configuring xTR	206

Example: Configuring MSMR	207
Example: Configuring RTR	208
Example: Verifying LISP Support for Disjoint RLOC Domains	208
Additional References for LISP Support for Disjoint RLOC Domains	210
Feature Information for LISP Support for Disjoint RLOC Domains	211

CHAPTER 11**LISP Data Plane Security 213**

Finding Feature Information	213
Prerequisites for LISP Data Plane Security	214
Restrictions for LISP Data Plane Security	214
Information About LISP Data Plane Security	214
Source RLOC Decapsulation Filtering	214
TCP-based Sessions for LISP Packet Transport	216
How to Configure LISP Data Plane Security	216
Configuring MSMR	216
Configuring the xTRs	218
Configuring PxTR	219
Verifying LISP Data Plane Security On a Map-Server	220
Verifying and Troubleshooting LISP Data Plane Security on an xTR or PxTR	221
Configuration Examples for LISP Data Plane Security	223
Example: Configuring MSMR	223
Example: Configuring the xTRs	223
Example: Configuring PxTR	223
Additional References for LISP Data Plane Security	224
Feature Information for LISP Data Plane Security	225

CHAPTER 12**LISP Reliable Registration 227**

Information About LISP Reliable Registration	227
LISP Reliable Map Registration	227
Verifying the LISP Reliable Registration	229
Additional References for LISP Reliable Registration	231
Feature Information for LISP Reliable Registration	231

CHAPTER 13**Overlapping Prefix 233**

Prerequisites for Overlapping Prefix	233
Information About Overlapping Prefix	233
Endpoint ID (EID)	233
EID-Prefix	233
Map Server/Map Resolver (MS/MR)	233
How to Configure Overlapping Prefix	234
Configuring Overlapping Prefix	234
Verifying Overlapping Prefix	234
Additional References for Overlapping Prefix	235
Feature Information for Overlapping Prefix	236

CHAPTER 14**LISP Generalized SMR 237**

Information About LISP Generalized SMR	237
Solicit-Map-Request (SMR)	237
Generalized SMR (GSMR)	237
Verifying LISP Generalized SMR	238
Additional References for LISP Reliable Registration	240
Feature Information for LISP Generalized SMR	241

CHAPTER 15**TTL Propagate Disable and Site-ID Qualification 243**

Information About TTL Propagate Disable and Site-ID Qualification	243
LISP Site	243
Map Server (MS)	243
Routing Locator (RLOC)	243
Traceroute Tool	243
Site ID Qualification	244
TTL Propagation	245
How to Configure Site ID Qualification	246
Configuring Site ID Qualification	246
Example: Site ID Qualification	246
How to Disable TTL Propagation	247
Disabling TTL Propagation for EID-Table	247
Disabling TTL Propagation for Router LISP Tag	247
Verifying TTL Propagate Disable	247

Additional References for TTL Propagate Disable and Site-ID Qualification	249
Feature Information for TTL Propagate Disable and Site-ID Qualification	249

CHAPTER 16

DNA SA Border Node Support	251
Finding Feature Information	251
Restrictions for DNA SA Border Node Support	251
Information About DNA SA Border Node Support	252
Enabling VxLAN Encapsulation for LISP Control Plane	252
Configuring Border Node as LISP PxTR	252
Configuring Border Node as LISP xTR	253
Security Group Tag (SGT) Propagation	255
Configuration Example: Border Node as LISP PxTR	255
Configuration Example: Border Node as LISP xTR	259
Feature Information for DNA SA Border Node Support	261



CHAPTER 1

Read Me First

Important Information about Cisco IOS XE 16

Effective Cisco IOS XE Release 3.7.0E for Catalyst Switching and Cisco IOS XE Release 3.17S (for Access and Edge Routing) the two releases evolve (merge) into a single version of converged release—the Cisco IOS XE 16—providing one release covering the extensive range of access and edge products in the Switching and Routing portfolio.

Feature Information

Use [Cisco Feature Navigator](#) to find information about feature support, platform support, and Cisco software image support. An account on Cisco.com is not required.

Related References

- [Cisco IOS Command References, All Releases](#)

Obtaining Documentation and Submitting a Service Request

- To receive timely, relevant information from Cisco, sign up at [Cisco Profile Manager](#).
- To get the business impact you're looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit [Cisco Marketplace](#).
- To obtain general networking, training, and certification titles, visit [Cisco Press](#).
- To find warranty information for a specific product or product family, access [Cisco Warranty Finder](#).



CHAPTER 2

Locator ID Separation Protocol (LISP) Overview

Locator ID Separation Protocol (LISP) is a network architecture and protocol that implements the use of two namespaces instead of a single IP address:

- Endpoint identifiers (EIDs)—assigned to end hosts.
- Routing locators (RLOCs)—assigned to devices (primarily routers) that make up the global routing system.

Splitting EID and RLOC functions yields several advantages including improved routing system scalability, and improved multihoming efficiency and ingress traffic engineering.

LISP functionality requires LISP-specific configuration of one or more LISP-related devices, such as the LISP egress tunnel router (ETR), ingress tunnel router (ITR), proxy ETR (PETR), proxy ITR (PITR), map resolver (MR), map server (MS), and LISP alternative logical topology (ALT) device.

- [Finding Feature Information, on page 3](#)
- [Prerequisites for Configuring LISP, on page 3](#)
- [Restrictions for Configuring LISP, on page 4](#)
- [Information About Configuring LISP, on page 4](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Configuring LISP

Before you can configure Locator/ID Separation Protocol (LISP), you will need to determine the type of LISP deployment you intend to deploy. The LISP deployment defines the necessary functionality of LISP devices, which, in turn, determines the hardware, software, and additional support from LISP mapping services and proxy services that are required to complete the deployment.

LISP configuration requires the data9 license.

Restrictions for Configuring LISP

- LISP is not supported on Tunnels.
- Management traffic generated on a LISP xTR with the source of a LISP EID interface does not work because management traffic such as SSH or telnet are not LISP aware. To make management protocols LISP aware, you need to create a static route pointing towards correct next hop. The static route should have a next hop of the LISP virtual interface and IP of the RLOC for the remote ETR. Management traffic generated on an xTR from a LISP EID interface needs this route inserted in the routing table as a workaround of this limitation.

Information About Configuring LISP

LISP Functionality Overview

Problem

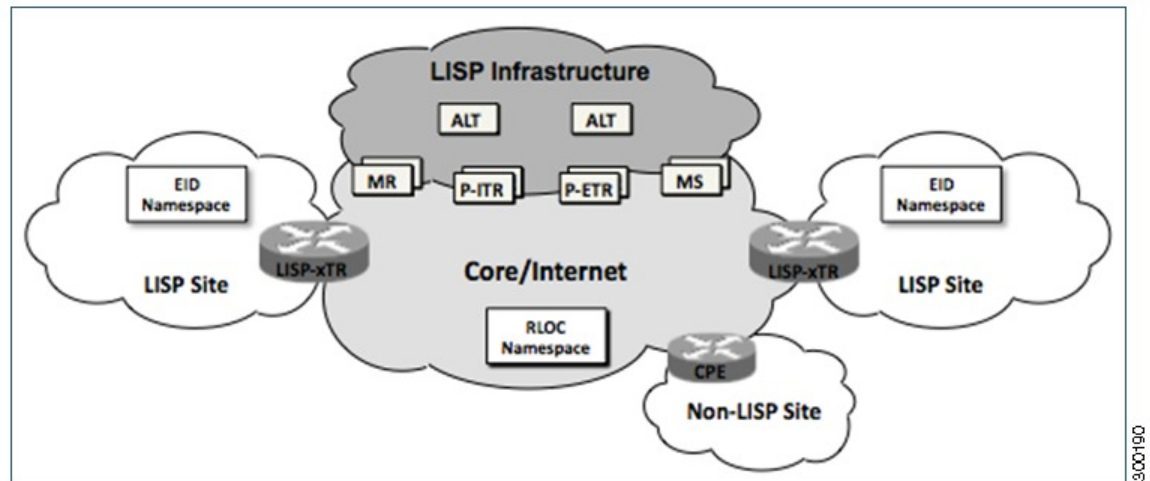
The continuous growth of the Internet presents a number of challenges. Among the most fundamental of these challenges is ensuring that the routing and addressing system continues to function efficiently even as the number of connected devices continues to increase. A basic observation during early network research and development work was that the single IP address, which includes both identity and location, leads to suboptimal route scaling and hinders multihoming and device mobility.

Solution

Locator ID Separation Protocol (LISP) provides improved routing scalability and facilitates flexible address assignment for multi-homing, provider independence, mobility, and virtualization. LISP offers an alternative to traditional Internet architecture by introducing two separate IP addresses: one to indicate routing locators (RLOCs) for routing traffic through the global Internet and a second address for endpoint identifiers (EIDs) used to identify network sessions between devices.

The figure below displays a general overview illustration of a LISP deployment environment, including the three essential environments that exist in a LISP environment: LISP sites (EID namespace), non-LISP sites (RLOC namespace), and LISP mapping service (infrastructure).

Figure 1: LISP Deployment Environment



As illustrated in the figure, the LISP EID namespace represents customer end sites in the same way that end sites are defined in non-LISP environments with one difference: The IP addresses used within these LISP sites are not advertised within the non-LISP Internet (RLOC namespace). Instead, end-customer LISP functionality is deployed exclusively on customer endpoint routers, which perform both the egress tunnel router (ETR) and ingress tunnel router (ITR) functions of a LISP device (abbreviated as xTR in the figure).

To fully implement LISP with support for mapping services and Internet interworking may require additional LISP infrastructure components as part of the deployment. As displayed in the figure above, these additional LISP infrastructure components include devices that function in the LISP roles of map resolver (MR), map server (MS), proxy egress tunnel router (PETR), proxy ingress tunnel router (PITR), and LISP alternative logical topology (ALT) device.

LISP Network Element Functions

The LISP architecture defines seven LISP-specific network infrastructure components. In some cases, a single physical device can implement more than one of these logical components. For more information, refer to the descriptions of the LISP components described in the following sections:

LISP Alternative Logical Topology

An alternative logical topology (ALT) device (not present in all mapping database deployments) connects through generic routing encapsulation (GRE) tunnels and border gateway protocol (BGP) sessions, map resolvers, map servers, and other ALT routers. The only purpose of ALT routers is to accept EID (Endpoint Identifier) prefixes advertised by devices that form a hierarchically distinct part of the EID numbering space and then advertise an aggregated EID prefix that represents that distinct space to other parts of the ALT. Just as in the global Internet routing system, this aggregation is performed to reduce the number of prefixes that need to be propagated throughout the entire network. An MS or combined MR/MS may also be configured to perform the functions of an ALT router.

LISP Egress Tunnel Router

An ETR connects a site to the LISP-capable part of a core network (such as the Internet), publishes EID-to-RLOC mappings for the site, responds to Map-Request messages, and decapsulates and delivers

LISP-encapsulated user data to end systems at the site. During operation, an ETR sends periodic Map-Register messages to all its configured map servers. The Map-Register messages contain all the EID-to-RLOC entries for the EID-numbered networks that are connected to the ETR's site.

An ETR that receives a Map-Request message verifies that the request matches an EID for which it is authoritative, constructs an appropriate Map-Reply message containing its configured mapping information, and sends this message to the ingress tunnel router (ITR) whose RLOCs are listed in the Map-Request message. An ETR that receives a LISP-encapsulated packet that is directed to one of its RLOCs decapsulates the packet, verifies that the inner header is destined for an EID-numbered end system at its site, and then forwards the packet to the end system using site-internal routing.

The ETR function is usually implemented in the customer premises equipment (CPE) router and does not require hardware changes on software-switched platforms, such as a Cisco Integrated Services Router (ISR). The same CPE router will often provide both ITR and ETR functions and, when doing so, is referred to as an xTR.

LISP Ingress Tunnel Router (ITR)

An ITR is responsible for finding EID-to-RLOC mappings for all traffic destined for LISP-capable sites. When the ITR receives a packet destined for an EID, it first looks for the EID in its mapping cache. If the ITR finds a match, it encapsulates the packet inside a LISP header with one of its RLOCs as the IP source address and one of the RLOCs from the mapping cache entry as the IP destination. The ITR then routes the packet normally.

If no entry is found in the ITR's mapping cache, the ITR sends a Map-Request message to one of its configured map resolvers and then discards the original packet. When the ITR receives a response to its Map-Request message, it creates a new mapping cache entry with the contents of the Map-Reply message. When another packet, such as a retransmission for the original and, now, discarded packet arrives, the new mapping cache entry is used for encapsulation and forwarding.



Note Sometimes the Map-Reply message will indicate that the destination is not an EID. When this happens, a negative mapping cache entry is created, which causes packets to either be discarded or forwarded natively when the packets match that cache entry.

Like the ETR, an ITR is usually implemented in a LISP site's customer premises equipment (CPE) router, which is typically configured as an xTR (performs functions of both ETR and ITR components).

LISP Map Resolver

Like an MS, a LISP MR connects to the ALT. The function of the LISP MR is to accept encapsulated Map-Request messages from ingress tunnel routers (ITRs), decapsulate those messages, and then forward the messages to the MS responsible for the egress tunnel routers (ETRs) that are authoritative for the requested EIDs.

When an MR is implemented concurrently with an MS in a private mapping system deployment, the concurrent MS forwards the encapsulated Map-Request messages to the authoritative ETRs. When a LISP ALT is present in the deployment, the MR forwards the Map-Request messages directly over the ALT to the MS responsible for the ETRs that are authoritative for the requested EIDs. An MR also sends Negative Map-Replies to ITRs in response to queries for non-LISP addresses.

LISP Map Server

An MS implements part of the distributed LISP mapping database by accepting registration requests from its client egress tunnel routers (ETRs), aggregating the successfully registered EID prefixes of those ETRs, and advertising the aggregated prefixes into the alternative logical topology (ALT) with border gateway protocol (BGP).

In a small private mapping system deployment, an MS may be configured to stand alone (or there may be several MSs) with all ETRs configured to register to each MS. If more than one, all MSs have full knowledge of the mapping system in a private deployment.

In a larger or public mapping system deployment, an MS is configured with a partial mesh of generic routing encapsulation (GRE) tunnels and BGP sessions to other map server systems or ALT routers. For these deployments, ETRs need to register to only one MS (or a few if redundancy is desired) and an ALT device is used to ensure that the entire LISP mapping system is available to all MS and MR devices.

Because an MS does not forward user data traffic—it handles only LISP control plane traffic—it does not require high performance switching capability and is well suited for implementation on a general purpose router, such as a Cisco Integrated Services Router (ISR). Both MS and MR functions are typically implemented on the same device, which is referred to as an MR/MS device.

LISP Proxy ETR

A LISP PETR implements ETR functions on behalf of non-LISP sites. A PETR is typically used when a LISP site needs to send traffic to non-LISP sites but the LISP site is connected through an access network of a service provider that does not accept nonroutable EIDs as packet sources.

When dual-stacked, a PETR may also serve as a way for EIDs and RLOCs to communicate in a LISP site that contains EIDs in one address family and RLOCs in a different address family. A dual-stacked PETR also provides multiaddress family support for LISP EIDs within one address family to be able to communicate with non-LISP destinations in the same address family over a core network within a different address family.

Example

A LISP site with IPv4-only RLOC connectivity can send IPv6 EIDs within an IPv4 LISP header across the IPv4 Internet to a dual-stacked PETR where the packets are decapsulated and then forwarded natively to non-LISP IPv6 Internet sites.

The PETR function is commonly configured on a device that also functions as a PITR. A device that functions as both a PETR and a PITR is known as a PxTR. Additionally, a PETR carries LISP data plane traffic and can be a high packet-rate device. To take advantage of this high packet-rate capability, deployments typically include hardware-switched platforms or high-end Cisco Integrated Services Routers (ISRs).

LISP Proxy ITR

A LISP PITR implements ITR mapping database lookups and LISP encapsulation functions on behalf of non-LISP-capable sites. PITRs are typically deployed near major Internet exchange points (IXPs) or in ISP networks to allow non-LISP customers from those networks to connect to LISP sites. In addition to implementing ITR functionality, a PITR also advertises some or all of the non-routable EID prefix space to the part of the non-LISP-capable Internet that it serves so that the non-LISP sites will route traffic toward the PITR for encapsulation and forwarding to LISP sites.



Note PITR advertising of nonroutable EID prefix space is intended to be highly aggregated with many EID prefixes represented by each prefix that is advertised by a PITR.

Like the PETR, when dual-stacked, the PITR also provides multiple-address family support. But the PITR supports transport of non-LISP traffic from one address family to LISP sites in the same address family over a core network within a different address family.

Example

A LISP site with IPv4-only RLOC connectivity can take advantage of a dual-stacked PITR to allow non-LISP IPv6 Internet users to reach IPv6 EIDs across the IPv4 Internet.

The PITR function is commonly configured on a device that also functions as a PETR. A device that functions as both a PETR and a PITR is known as a PxTR. Additionally, a PITR carries LISP data plane traffic and can be a high packet-rate device. To take advantage of this high packet-rate capability, deployments typically include hardware-switched platforms or high-end Cisco® Integrated Services Routers (ISRs).

Feature Information for LISP Overview

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for LISP Overview

Feature Name	Releases	Feature Information
LISP Overview	15.1(4)M Cisco IOS XE Release 3.3.0S	The LISP Overview feature provides a general overview of LISP and its components. The following LISP components are supported: <ul style="list-style-type: none"> • Egress tunnel router (ETR) • Ingress tunnel router (ITR) • LISP alternative logical topology (ALT) device • Map resolver (MR) • Map server (MS) • Proxy ETR (PETR) • Proxy ITR (PITR)

Feature Name	Releases	Feature Information
LISP, SHA-2 support for site registration	15.3(2)T Cisco IOS XE Release 3.9S	<p>LISP can be configured to use SHA2-based HMAC algorithm for integrity-checking LISP site registration messages. Prior to this release, only SHA1-based HMAC algorithm was supported.</p> <p>The following commands were modified:</p> <ul style="list-style-type: none">• ipv4 etr map-server• ipv6 etr map-server



CHAPTER 3

Configuring LISP (Locator ID Separation Protocol)

This guide describes how to configure basic Locator ID Separation Protocol (LISP) functionality on all LISP-related devices, including the egress tunnel router (ETR), ingress tunnel router (ITR), proxy ETR (PETR), proxy ITR (PITR), map resolver (MR), and map server (MS).

LISP is a network architecture and protocol that implements the use of two namespaces instead of a single IP address. These namespaces, known as endpoint identifiers (EIDs), are assigned to end-hosts and routing locators (RLOCs), which are assigned to devices (primarily routers) that make up the global routing system. Splitting EID and RLOC functions delivers improvements in routing system scalability, multi-homing efficiency, and ingress traffic engineering.

- [Prerequisites for Configuring LISP, on page 11](#)
- [How to Configure LISP, on page 11](#)
- [Additional References for Configuring LISP, on page 84](#)
- [Feature Information for LISP, on page 85](#)

Prerequisites for Configuring LISP

- If a LISP xTR is also a First Hop Router (FH) or a Rendezvous Point (RP), then the xTR needs to have at least one connected interface that is covered by a local LISP database mapping. Before an ITR forwards traffic over LISP, it does a source check to ensure that the source address of the traffic stream is a local EID (database mapping). Since PIM register and register-stop messages are sourced directly from the router itself, to be forwarded over LISP, the messages must come from an interface covered by a database mapping. A loopback or other connected interface is fine for this purpose. No additional configuration is required to ensure the proper address is selected.

This prerequisite is not required on a Proxy xTR, which does not do a source check.

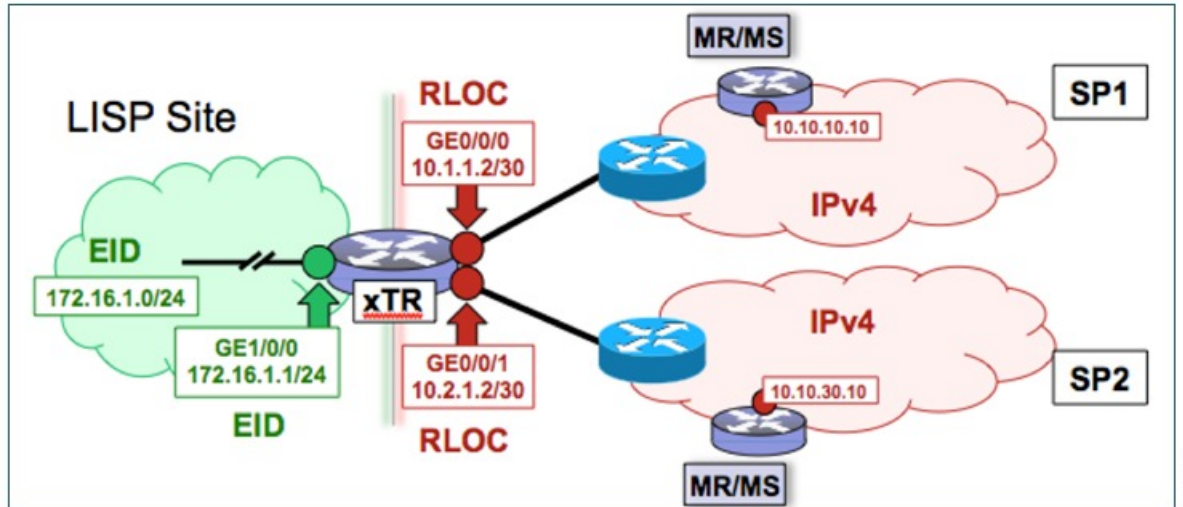
How to Configure LISP

Configure a Dual-Homed LISP Site with Two IPv4 RLOCs and an IPv4 EID

Perform this task to configure a dual-homed LISP site with two IPv4 RLOCs and an IPv4 EID. In this task, a LISP site uses a single edge router configured as both an ITR and an ETR (known as an xTR) with two connections to upstream providers. Both of the RLOCs and the EID prefix are IPv4. The LISP site registers

to two map resolver/map server (MR/MS) devices in the network core. The topology used in this LISP configuration is shown in the figure below.

Figure 2: Dual-Homed LISP Site with Two IPv4 RLOCs and an IPv4 EID



The components illustrated in the topology shown in the figure are described below:

• **LISP site:**

- The CPE functions as a LISP ITR and ETR (xTR).
- The LISP xTR is authoritative for the IPv4 EID prefix of 172.16.1.0/24.
- The LISP xTR has two RLOC connections to the core. The RLOC connection to SP1 is 10.1.1.2/30; the RLOC connection to SP2 is 10.2.1.2/30.
- For this simple dual-homed configuration, the LISP site policy specifies equal load sharing between service provider (SP) links for ingress traffic engineering.

• **Mapping system:**

- Two map resolver/map server (MR/MS) systems are assumed to be available for the LISP xTR to configure. The MR/MSs have IPv4 RLOCs 10.10.10.10 and 10.10.30.10.
- Mapping Services are assumed to be provided as part of this LISP solution via a private mapping system or as a public LISP mapping system. From the perspective of the configuration of this LISP site xTR, there is no difference.



Note Map server and map resolver configurations are not shown here. See the "Configure a Private LISP Mapping System Using a Standalone Map Resolver/Map Server" section for information about map server and map resolver configuration.

This task shows how to enable and configure LISP ITR and ETR (xTR) functionality when using a LISP map server and map resolver for mapping services.

SUMMARY STEPS

1. **configure terminal**
2. **router lisp**
3. Do one of the following:
 - **database-mapping** *EID-prefix/prefix-length locator priority priority weight weight*
 - **database-mapping** *EID-prefix/prefix-length ipv4-interface locator priority priority weight weight*
4. Repeat one of the choices in Step 3 to configure a second RLOC.
5. **ipv4 itr**
6. **ipv4 etr**
7. **ipv4 itr map-resolver** *map-resolver-address*
8. **ipv4 etr map-server** *map-server-address key key-type authentication-key*
9. **exit**
10. **ip route** *ipv4-prefix next-hop*
11. **exit**

DETAILED STEPS

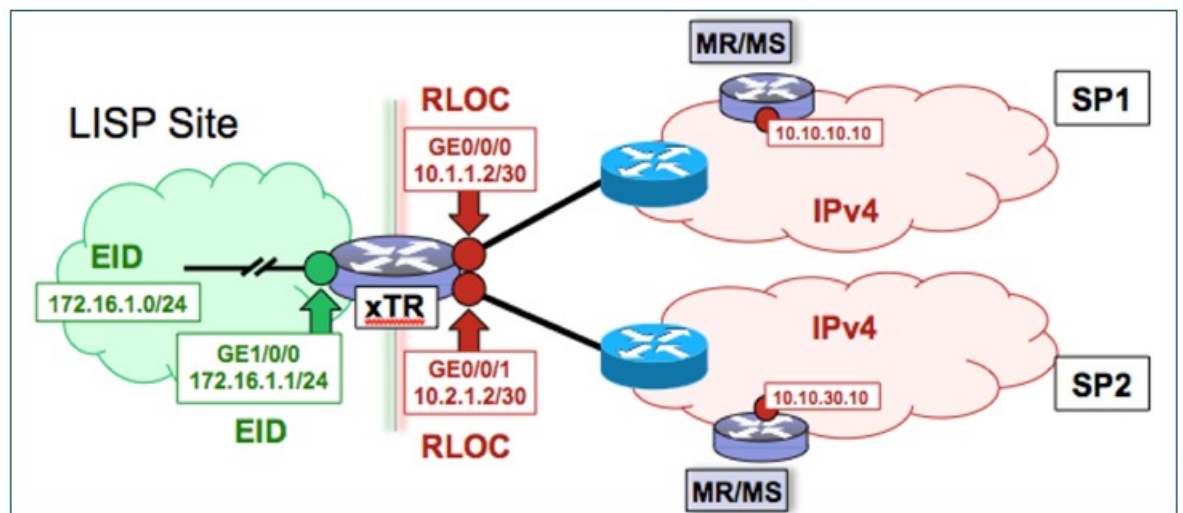
	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 2	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 3	Do one of the following: <ul style="list-style-type: none"> • database-mapping <i>EID-prefix/prefix-length locator priority priority weight weight</i> • database-mapping <i>EID-prefix/prefix-length ipv4-interface locator priority priority weight weight</i> Example: <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50</pre> Example: <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 ipv4-interface GigabitEthernet0/0/0 priority 1 weight 50</pre>	Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site. <ul style="list-style-type: none"> • In this step example, a single EID prefix, 172.16.1.0/24, is being associated with the single IPv4 RLOC 10.1.1.2 but the <i>weight</i> argument of 50 signifies that a second database-mapping command is to be configured in the next step. • In the second example, the configuration shows the use of the dynamic interface form of the database-mapping command. This form is useful when the RLOC address is obtained dynamically, such as via DHCP.
Step 4	Repeat one of the choices in Step 3 to configure a second RLOC.	—

	Command or Action	Purpose
Step 5	ipv4 itr Example: <pre>Router(config-router-lisp)# ipv4 itr</pre>	Enables LISP ITR functionality for the IPv4 address family.
Step 6	ipv4 etr Example: <pre>Router(config-router-lisp)# ipv4 etr</pre>	Enables LISP ETR functionality for the IPv4 address family.
Step 7	ipv4 itr map-resolver map-resolver-address Example: <pre>Router(config-router-lisp)# ipv4 itr map-resolver 10.10.10.10</pre>	<p>Configures the locator address of the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 8	ipv4 etr map-server map-server-address key key-type authentication-key Example: <pre>Router(config-router-lisp)# ipv4 etr map-server 10.10.10.10 key 0 some-key</pre>	<p>Configures the locator address of the LISP map server and the authentication key that this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> The map server must be configured with EID prefixes matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map server is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.)</p> <p>Note Up to two map servers may be configured if multiple map servers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 9	exit Example: <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.

	Command or Action	Purpose
Step 10	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> LISP-encapsulated to a LISP site when traffic is LISP-to-LISP natively forwarded when traffic is LISP-to-non-LISP. Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP EID and the destination matches one of the following entries: <ul style="list-style-type: none"> a current map-cache entry a default route with a legitimate next-hop no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
Step 11	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	<p>Exits global configuration mode.</p>

Example:

Figure 3: Dual-Homed LISP Site with Two IPv4 RLOCs and an IPv4 EID



This example shows the complete configuration for the LISP topology illustrated in the figure above and in this task.

```

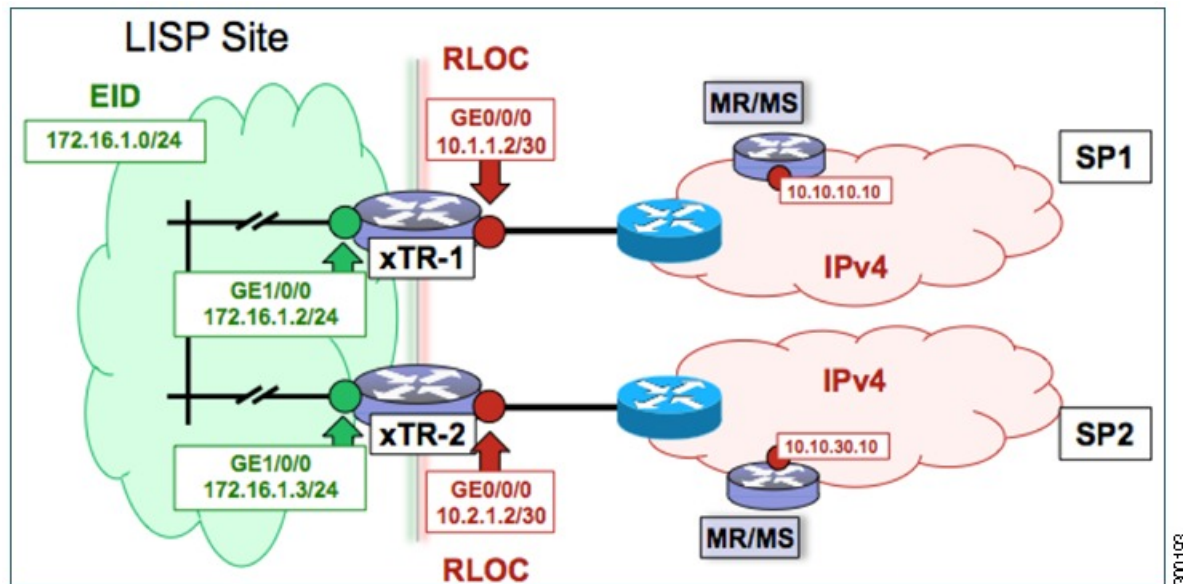
hostname xTR
!
no ip domain lookup
ip cef
!
interface Loopback0
 ip address 172.17.1.1 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
 description Link to SP1 (RLOC)
 ip address 10.1.1.2 255.255.255.252
!
interface GigabitEthernet0/0/1
 description Link to SP2 (RLOC)
 ip address 10.2.1.2 255.255.255.252
!
interface GigabitEthernet1/0/0
 description Link to Site (EID)
 ip address 172.16.1.1 255.255.255.0
!
router lisp
 database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
 database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
 ipv4 itr
 ipv4 etr
 ipv4 itr map-resolver 10.10.10.10
 ipv4 itr map-resolver 10.10.30.10
 ipv4 etr map-server 10.10.10.10 key 0 some-key
 ipv4 etr map-server 10.10.30.10 key 0 some-key
 exit
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
ip route 0.0.0.0 0.0.0.0 10.2.1.1

```

Configure a Multihomed LISP Site with Two xTRs and Two IPv4 RLOCs and an IPv4 EID

Perform this task to configure a multihomed LISP site with two xTRs, two IPv4 RLOCs, and an IPv4 EID. In this task, a LISP site uses two edge routers. Each edge router is configured as an xTR (each performs as both an ITR and an ETR) and each also includes a single IPv4 connection to an upstream provider. (Two different providers are used in this example but the same upstream provider could be used for both connections.) Both of the RLOCs and the EID prefix are IPv4. The LISP site registers to two map resolver/map server (MR/MS) devices in the network core. The topology used in this typical multihomed LISP configuration is shown in the figure below.

Figure 4: Typical Multihomed LISP Site with Two xTRs and Two IPv4 RLOCs and an IPv4 EID



The components illustrated in the topology shown in the figure are described below:

• **LISP site:**

- Two CPE routers make up the LISP site: xTR-1 and xTR-2.
- Both CPE routers function as LISP xTRs (that is, an ITR and an ETR).
- The LISP site is authoritative for the IPv4 EID prefix of 172.16.1.0/24.
- Each LISP xTR has a single IPv4 RLOC connection to the core: the RLOC connection for xTR-1 to SP1 is 10.1.1.2/30; the RLOC connection for xTR-2 to SP2 is 10.2.1.2/30.
- For this multihomed case, the LISP site policy specifies equal load-sharing between service provider (SP) links for ingress traffic engineering.

• **Mapping system:**

- Two map resolver/map server (MR/MS) systems are assumed to be available for the LISP xTR to configure. The MR/MSs have IPv4 RLOCs 10.10.10.10 and 10.10.30.10.
- Mapping services are assumed to be provided as part of this LISP solution via a private mapping system or as a public LISP mapping system. From the perspective of the configuration of these LISP site xTRs, there is no difference.



Note Map server and map resolver configurations are not shown here. See the "Configure a Private LISP Mapping System Using a Standalone Map Resolver/Map Server" section for information about map server and map resolver configuration.

Perform the steps in this task (once through for each xTR in the LISP site) to enable and configure LISP ITR and ETR (xTR) functionality when using a LISP map server and map resolver for mapping services. The

example configurations at the end of this task show the full configuration for configuring two xTRs (xTR1 and xTR2).

SUMMARY STEPS

1. **configure terminal**
2. **router lisp**
3. **database-mapping** *EID-prefix/prefix-length locator priority priority weight weight*
4. Repeat Step 3 to configure a second RLOC for the same xTR.
5. **ipv4 itr**
6. **ipv4 etr**
7. **ipv4 itr map-resolver** *map-resolver-address*
8. Repeat Step 7 to configure a second locator address for the map resolver.
9. **ipv4 etr map-server** *map-server-address key key-type authentication-key*
10. Repeat Step 9 to configure a second locator address for the map server.
11. **exit**
12. **ip route** *ipv4-prefix next-hop*
13. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 2	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 3	database-mapping <i>EID-prefix/prefix-length locator priority priority weight weight</i> Example: <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50</pre>	Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site. <ul style="list-style-type: none"> • In this step example, a single EID prefix, 172.16.1.0/24, is being associated with a LISP site that contains two separate xTRs. Each xTR has a single IPv4 RLOC connection to the core. In this example, xTR-1 has an IPv4 RLOC connection to SP1 at 10.1.1.2 but the <i>weight</i> argument of 50 signifies that a second database-mapping command is to be configured in the next step.

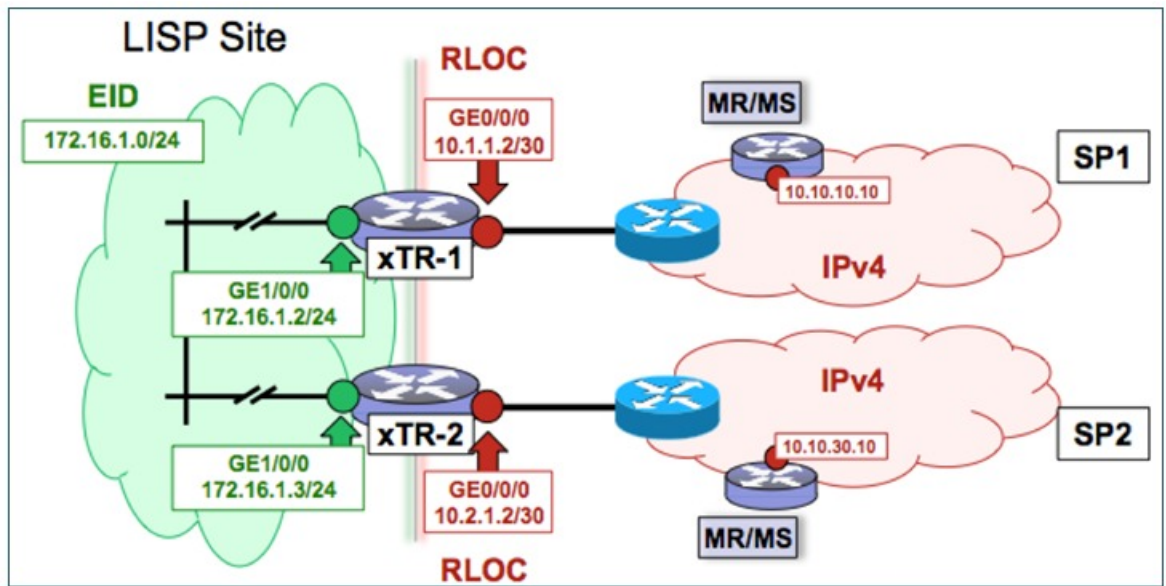
	Command or Action	Purpose
		<p>Note Two database-mapping commands are required on each xTR to indicate to the mapping system that this LISP site is reachable via these two IPv4 RLOCs. In this example, one RLOC is local (connected) to one xTR and the other is local (connected) to the other xTR.</p>
Step 4	<p>Repeat Step 3 to configure a second RLOC for the same xTR.</p> <p>Example:</p> <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50</pre>	<p>Configures an EID-to-RLOC mapping relationship and its associated traffic policy for an xTR on this LISP site.</p> <ul style="list-style-type: none"> In this step example, the second RLOC connection for xTR-1 has an IPv4 RLOC connection to SP2 (10.2.1.2). <p>Note When a LISP site contains multiple xTRs, all xTRs must be configured with identical database-mapping commands to provide the mapping system with consistent information about EID-to-RLOC mappings.</p>
Step 5	<p>ipv4 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr</pre>	Enables LISP ITR functionality for the IPv4 address family.
Step 6	<p>ipv4 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr</pre>	Enables LISP ETR functionality for the IPv4 address family.
Step 7	<p>ipv4 itr map-resolver map-resolver-address</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 10.10.10.10</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 8	<p>Repeat Step 7 to configure a second locator address for the map resolver.</p> <p>Example:</p>	Configures a second locator address for the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions.

	Command or Action	Purpose
	Router(config-router-lisp)# ipv4 itr map-resolver 10.10.30.10	
Step 9	<p>ipv4 etr map-server <i>map-server-address key key-type authentication-key</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr map-server 10.10.10.10 key 0 some-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key that this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> • In this example, each xTR must register to both map servers. • The map server must be configured with EID prefixes matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map server is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.)</p> <p>Note Up to two map servers may be configured if multiple map servers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 10	<p>Repeat Step 9 to configure a second locator address for the map server.</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr map-server 10.10.30.10 key 0 some-key</pre>	<p>Configures a second locator address for the LISP map server and the authentication key that this router will use to register with the LISP mapping system.</p>
Step 11	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	<p>Exits LISP configuration mode and returns to global configuration mode.</p>
Step 12	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> • All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> • LISP-encapsulated to a LISP site when traffic is LISP-to-LISP • natively forwarded when traffic is LISP-to-non-LISP • Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP

	Command or Action	Purpose
		<p>EID and the destination matches one of the following entries:</p> <ul style="list-style-type: none"> • a current map-cache entry • a default route with a legitimate next-hop • no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
<p>Step 13</p>	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	<p>Exits global configuration mode.</p>

Example:

Figure 5: Typical Multihomed LISP Site with Two xTRs and Two IPv4 RLOCs and an IPv4 EID



The examples below show the complete configuration for the LISP topology illustrated in the figure above and in this task:

Example configuration for xTR-1:

```
!
hostname xTR-1
!
no ip domain lookup
ip cef
!
```

```

interface Loopback0
 ip address 172.17.1.1 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
 description Link to SP1 (RLOC)
 ip address 10.1.1.2 255.255.255.252
!
interface GigabitEthernet1/0/0
 description Link to Site (EID)
 ip address 172.16.1.2 255.255.255.0
!
router lisp
 database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
 database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
 ipv4 itr
 ipv4 etr
 ipv4 itr map-resolver 10.10.10.10
 ipv4 itr map-resolver 10.10.30.10
 ipv4 etr map-server 10.10.10.10 key 0 some-key
 ipv4 etr map-server 10.10.30.10 key 0 some-key
 exit
!
 ip route 0.0.0.0 0.0.0.0 10.1.1.1

```

Example configuration for xTR-2:

```

!
hostname xTR-2
!
no ip domain lookup
ip cef
!
interface Loopback0
 ip address 172.17.1.2 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
 description Link to SP2 (RLOC)
 ip address 10.2.1.2 255.255.255.252
!
interface GigabitEthernet1/0/0
 description Link to Site (EID)
 ip address 172.16.1.3 255.255.255.0
!
router lisp
 database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
 database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
 ipv4 itr
 ipv4 etr
 ipv4 itr map-resolver 10.10.10.10
 ipv4 itr map-resolver 10.10.30.10
 ipv4 etr map-server 10.10.10.10 key 0 some-key
 ipv4 etr map-server 10.10.30.10 key 0 some-key
 exit
!
 ip route 0.0.0.0 0.0.0.0 10.2.1.1

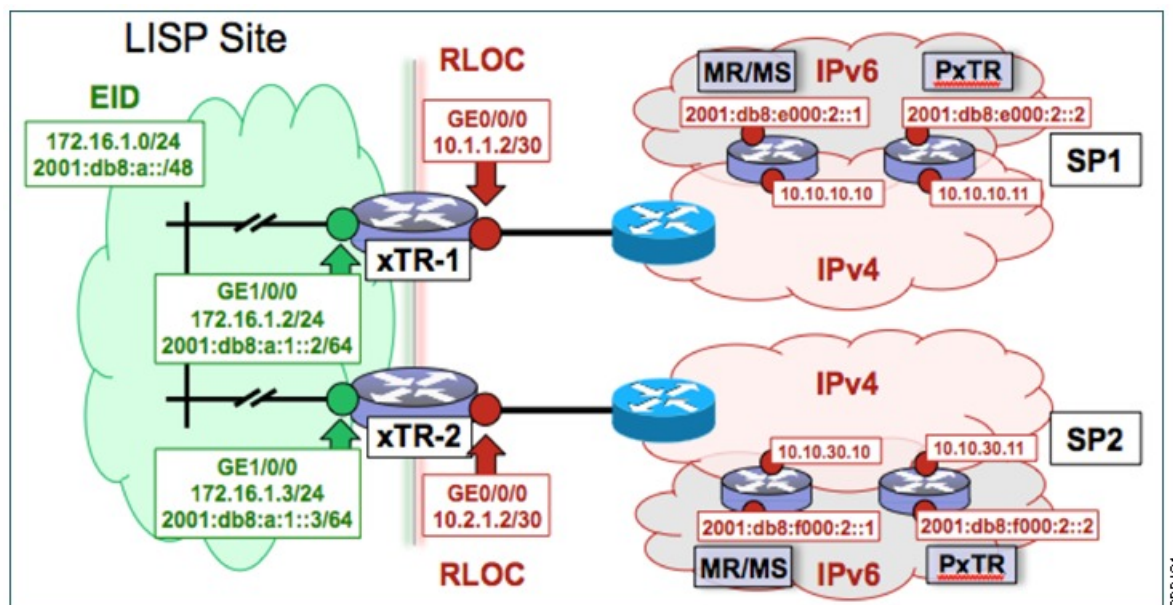
```


Configure a Multihomed LISP Site with Two xTRs and Two IPv4 RLOCs and Both an IPv4 and an IPv6 EID

Perform this task to configure a multihomed LISP site with two xTRs, two IPv4 RLOCs, and both an IPv4 and an IPv6 EID. In this task, a LISP site uses two edge routers. Each edge router is configured as an xTR (each performs as both an ITR and an ETR) and each also includes a single IPv4 connection to an upstream provider. (Two different providers are used in this example but the same upstream provider could be used for both connections.) Both of the RLOCs and one of the EIDs are IPv4. However, in this example, the LISP site includes an IPv6 EID, as well.

This LISP site requires the use of Proxy Ingress/Egress Tunnel Router (PxTR) LISP infrastructure for access to non-LISP IPv6 addresses. That is, the LISP site uses only its IPv4 RLOCs to reach IPv6 LISP and non-LISP addresses. Additionally, this LISP site registers to two map resolver/map server (MR/MS) devices in the network core. The topology used in this multihomed LISP configuration is shown in the figure below.

Figure 6: Multihomed LISP Site with Two xTRs, Two IPv4 RLOCs, and Both an IPv4 and an IPv6 EID



The components illustrated in the topology shown in the figure are described below:

- **LISP site:**
 - Two CPE routers make up the LISP site: xTR-1 and xTR-2.
 - Both CPE routers function as LISP xTRs (that is, an ITR and an ETR).
 - The LISP site is authoritative for both the IPv4 EID prefix of 172.16.1.0/24 and the IPv6 EID prefix 2001:db8:a::/48.
 - Each LISP xTR has a single RLOC connection to the core: the RLOC connection for xTR-1 to SP1 is 10.1.1.2/30; the RLOC connection for xTR-2 to SP2 is 10.2.1.2/30.
 - For this multihomed case, the LISP site policy specifies equal load-sharing between service provider (SP) links for ingress traffic engineering.

- **Mapping system:**

- Two map resolver/map server (MR/MS) systems are assumed to be available for the LISP xTR to configure. The MR/MSs have IPv4 RLOCs 10.10.10.10 and 10.10.30.10.
- Mapping services are assumed to be provided as part of this LISP solution via a private mapping system or as a public LISP mapping system. From the perspective of the configuration of these LISP site xTRs, there is no difference.



Note Map server and map resolver configurations are not shown here. See the "Configure a Private LISP Mapping System Using a Standalone Map Resolver/Map Server" section for information about map server and map resolver configuration.

- PxTR services are also assumed to be provided as part of this LISP solution via a private or public mapping system. From the perspective of the configuration of these LISP site xTRs, there is no difference.
- The PxTRs have IPv4 RLOCs of 10.10.10.11 and 10.10.30.11 and will be used (as PETRs) for LISP IPv6 EIDs to reach non-LISP IPv6 sites. Return traffic is attracted by the PITR function (with the assumption that the PITR advertises coarse aggregates for IPv6 LISP EIDs into the IPv6 core.)

Perform the steps in this task (once through for each xTR in the LISP site) to enable and configure LISP ITR and ETR (xTR) functionality when using a LISP map server and map resolver for mapping services. The example configurations at the end of this task show the full configuration for two xTRs (xTR1 and xTR2).

SUMMARY STEPS

1. **configure terminal**
2. **router lisp**
3. **database-mapping** *EID-prefix/prefix-length locator priority priority weight weight*
4. Repeat Step 3 to configure a second RLOC (10.2.1.2) for the same xTR and IPv4 EID prefix.
5. Repeat Step 3 and Step 4 to configure the same RLOC connections, again, for the same xTR but, when repeating these two steps, associate the IPv6 EID prefix, 2001:db8:a::/48, instead of the IPv4 EID prefix.
6. **ipv4 itr**
7. **ipv4 etr**
8. **ipv4 itr map-resolver** *map-resolver-address*
9. Repeat Step 8 to configure a second locator address of the map resolver.
10. **ipv4 etr map-server** *map-server-address key key-type authentication-key*
11. Repeat Step 10 to configure a second locator address for the map server.
12. **ipv6 itr**
13. **ipv6 etr**
14. **ipv6 itr map-resolver** *map-resolver-address*
15. Repeat Step 14 to configure a second locator address for the map resolver.
16. **ipv6 etr map-server** *map-server-address key key-type authentication-key*
17. Repeat Step 16 to configure a second locator address for the map server.
18. **ipv6 use-petr** *petr-address*
19. Repeat Step 18 to configure a second locator address for the PETR.

20. `exit`
21. `ip route ipv4-prefix next-hop`
22. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>router lisp</p> <p>Example:</p> <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 3	<p>database-mapping EID-prefix/prefix-length locator priority priority weight weight</p> <p>Example:</p> <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50</pre>	<p>Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site.</p> <ul style="list-style-type: none"> In steps 3, 4, and 5 of this example, an IPv4 EID prefix, 172.16.1.0/24, and an IPv6 prefix, 2001:db8:a::/48, are being associated with a LISP site that contains two separate xTRs that each have a single IPv4 RLOC connection to the core. In this first step example, xTR-1 is configured with an IPv4 RLOC connection to SP1 at 10.1.1.2 but the <i>weight</i> argument of 50 signifies that a second database-mapping command is to be configured in the next step. <p>Note Four database-mapping commands are required for each xTR to indicate to the mapping system that both the associated IPv4 and IPv6 EID prefixes are reachable at this LISP site via these two IPv4 RLOCs. In this example, one RLOC is local (connected) to one xTR and the other is local (connected) to the other xTR.</p>
Step 4	<p>Repeat Step 3 to configure a second RLOC (10.2.1.2) for the same xTR and IPv4 EID prefix.</p> <p>Example:</p> <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50</pre>	<p>Configures an EID-to-RLOC mapping relationship and its associated traffic policy for an xTR on this LISP site.</p> <ul style="list-style-type: none"> In this step example, the second RLOC connection for xTR-1 has an IPv4 RLOC connection to SP2 (10.2.1.2). <p>Note When a LISP site contains multiple xTRs, all xTRs must be configured with identical database-mapping commands to provide the mapping system with consistent information about EID-to-RLOC mappings.</p>

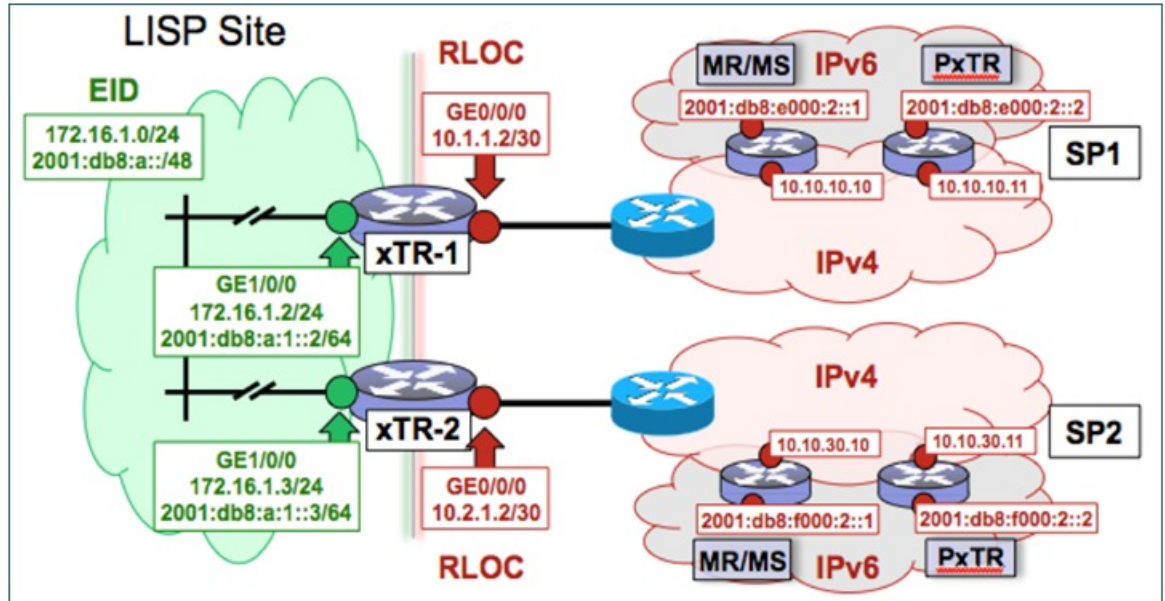
Configure a Multihomed LISP Site with Two xTRs and Two IPv4 RLOCs and Both an IPv4 and an IPv6 EID

	Command or Action	Purpose
Step 5	Repeat Step 3 and Step 4 to configure the same RLOC connections, again, for the same xTR but, when repeating these two steps, associate the IPv6 EID prefix, 2001:db8:a::/48, instead of the IPv4 EID prefix.	—
Step 6	ipv4 itr Example: Router(config-router-lisp)# ipv4 itr	Enables LISP ITR functionality for the IPv4 address family.
Step 7	ipv4 etr Example: Router(config-router-lisp)# ipv4 etr	Enables LISP ETR functionality for the IPv4 address family.
Step 8	ipv4 itr map-resolver map-resolver-address Example: Router(config-router-lisp)# ipv4 itr map-resolver 10.10.10.10	Configures a locator address for the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions. <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 9	Repeat Step 8 to configure a second locator address of the map resolver. Example: Router(config-router-lisp)# ipv4 itr map-resolver 10.10.30.10	Configures a second locator address for the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions.
Step 10	ipv4 etr map-server map-server-address key key-type authentication-key Example: Router(config-router-lisp)# ipv4 etr map-server 10.10.10.10 key 0 some-key	Configures a locator address for the LISP map server and an authentication key that this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system. <ul style="list-style-type: none"> In this example, each xTR must register to both map servers. The map server must be configured with EID prefixes matching those configured on this ETR and with an identical authentication key.

	Command or Action	Purpose
		<p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map server is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.)</p> <p>Note Up to two map servers may be configured if multiple map servers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 11	<p>Repeat Step 10 to configure a second locator address for the map server.</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr map-server 10.10.30.10 key 0 some-key</pre>	Configures a second locator address for the LISP map server and the authentication key that this router will use to register with the LISP mapping system.
Step 12	<p>ipv6 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr</pre>	Enables LISP ITR functionality for the IPv6 address family.
Step 13	<p>ipv6 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 etr</pre>	Enables LISP ETR functionality for the IPv6 address family.
Step 14	<p>ipv6 itr map-resolver map-resolver-address</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr map-resolver 10.10.10.10</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send Map-Request messages for IPv6 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 15	<p>Repeat Step 14 to configure a second locator address for the map resolver.</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr map-resolver 10.10.30.10</pre>	Configures a second locator address for the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions.
Step 16	<p>ipv6 etr map-server map-server-address key key-type authentication-key</p> <p>Example:</p>	Configures a locator address for the LISP map server and an authentication key that this router, acting as an IPv6 LISP ETR, will use to register to the LISP mapping system.

	Command or Action	Purpose
	<pre>Router(config-router-lisp)# ipv6 etr map-server 10.10.10.10 key 0 some-key</pre>	<ul style="list-style-type: none"> In this example, each xTR must register to both map servers. The map server must be configured with EID prefixes matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map server is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.)</p> <p>Note Up to two map servers may be configured if multiple map servers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 17	<p>Repeat Step 16 to configure a second locator address for the map server.</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr map-server 10.10.30.10 key 0 some-key</pre>	Configures a second locator address for the LISP map server and an authentication key that this router, acting as an IPv6 LISP ETR, will use to register with the LISP mapping system.
Step 18	<p>ipv6 use-petr petr-address</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 use-petr 10.10.10.11</pre>	<p>Configures a locator address for the Proxy Egress Tunnel Router (PETR) to which each xTR will forward LISP-encapsulated IPv6 EIDs (using the xTR's IPv4 RLOC) to reach non-LISP IPv6 addresses.</p> <p>Note The PETR is assumed to be dual-stacked and capable of natively reaching the non-LISP IPv6 address. In addition, the PITR is assumed to be dual-stacked and to be advertising coarse aggregates for IPv6 LISP EIDs into the IPv6 core to handle return traffic (non-LISP IPv6 to LISP IPv6 over an IPv4 infrastructure).</p> <p>Note The locator address of the PETR may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the PETR is reachable via its IPv4 locator address. (See the <i>LISP Command Reference</i> for more details.)</p> <p>Note Up to eight PETRs may be configured if multiple PETRs are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 19	<p>Repeat Step 18 to configure a second locator address for the PETR.</p>	Configures a second locator address for the PETR to which each xTR will forward LISP-encapsulated IPv6 EIDs

	Command or Action	Purpose
	<p>Example:</p> <pre>Router(config-router-lisp)# ipv6 use-petr 10.10.30.11</pre>	(using the xTR's IPv4 RLOC) to reach non-LISP IPv6 addresses.
Step 20	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 21	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> • All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> • LISP-encapsulated to a LISP site when traffic is LISP-to-LISP • natively forwarded when traffic is LISP-to-non-LISP • Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP EID and the destination matches one of the following entries: <ul style="list-style-type: none"> • a current map-cache entry • a default route with a legitimate next-hop • no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
Step 22	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	Exits global configuration mode.

Example:**Figure 7: Multihomed LISP Site with Two xTRs, Two IPv4 RLOCs, and Both an IPv4 and an IPv6 EID**

The examples below show the complete configuration for the LISP topology illustrated in the figure above and in this task:

Example configuration for xTR-1:

```

!
hostname xTR-1
!
no ip domain lookup
ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 ip address 172.17.1.1 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
 description Link to SP1 (RLOC)
 ip address 10.1.1.2 255.255.255.252
!
interface GigabitEthernet1/0/0
 description Link to Site (EID)
 ip address 172.16.1.2 255.255.255.0
 ipv6 address 2001:db8:a:1::2/64
!
router lisp
 database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
 database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
 database-mapping 2001:db8:a::/48 10.1.1.2 priority 1 weight 50
 database-mapping 2001:db8:a::/48 10.2.1.2 priority 1 weight 50
 ipv4 itr

```



```

ipv4 etr
ipv4 itr map-resolver 10.10.10.10
ipv4 itr map-resolver 10.10.30.10
ipv4 etr map-server 10.10.10.10 key 0 some-key
ipv4 etr map-server 10.10.30.10 key 0 some-key
ipv6 itr
ipv6 etr
ipv6 itr map-resolver 10.10.10.10
ipv6 itr map-resolver 10.10.30.10
ipv6 etr map-server 10.10.10.10 key 0 some-key
ipv6 etr map-server 10.10.30.10 key 0 some-key
ipv6 use-petr 10.10.10.11
ipv6 use-petr 10.10.30.11
exit
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
!
ipv6 route ::/0

```

Example configuration for xTR-2:

```

!
no ip domain lookup
ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 ip address 172.17.1.2 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
 description Link to SP2 (RLOC)
 ip address 10.2.1.2 255.255.255.252
!
interface GigabitEthernet1/0/0
 description Link to Site (EID)
 ip address 172.16.1.3 255.255.255.0
 ipv6 address 2001:db8:a::1::3/64
!
router lisp
 database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
 database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
 database-mapping 2001:db8:a::/48 10.1.1.2 priority 1 weight 50
 database-mapping 2001:db8:a::/48 10.2.1.2 priority 1 weight 50
 ipv4 itr
 ipv4 etr
 ipv4 itr map-resolver 10.10.10.10
 ipv4 itr map-resolver 10.10.30.10
 ipv4 etr map-server 10.10.10.10 key 0 some-xtr-key
 ipv4 etr map-server 10.10.30.10 key 0 some-xtr-key
 ipv6 itr
 ipv6 etr
 ipv6 itr map-resolver 10.10.10.10
 ipv6 itr map-resolver 10.10.30.10
 ipv6 etr map-server 10.10.10.10 key 0 some-xtr-key
 ipv6 etr map-server 10.10.30.10 key 0 some-xtr-key
 ipv6 use-petr 10.10.10.11
 ipv6 use-petr 10.10.30.11
 exit
!
ip route 0.0.0.0 0.0.0.0 10.2.1.1

```

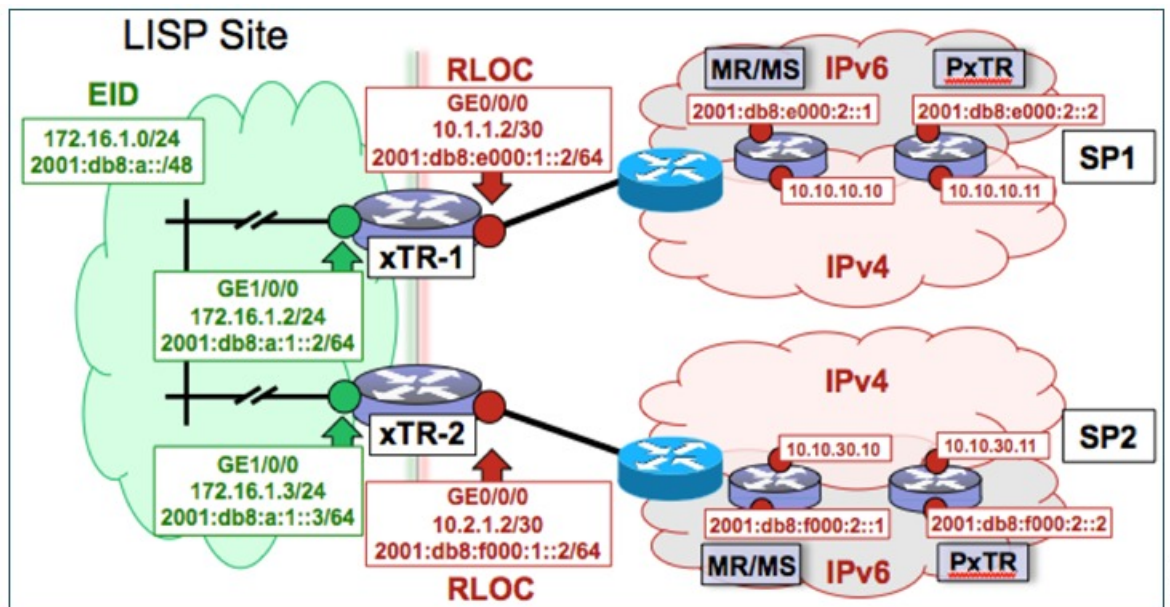
```
!
ipv6 route :::/0
```

Configure a Multihomed LISP Site with Two xTRs that Each have Both an IPv4 and an IPv6 RLOC and Both an IPv4 and an IPv6 EID

Perform this task to configure a multihomed LISP site with two xTRs, each with both an IPv4 and an IPv6 RLOC and both with an IPv4 and an IPv6 EID. In this task, a LISP site uses two edge routers. Each edge router is configured as an xTR (each performs as both an ITR and an ETR) and each also includes a single, dual stack (IPv4 and IPv6) connection to an upstream provider. (Two different providers are used in this example but the same upstream provider could be used for both connections.) Each xTR has an IPv4 RLOC and an IPv6 RLOC and both IPv4 and IPv6 EID prefixes are being used within the LISP site. However, because the site has both IPv4 and IPv6 RLOCs, it does not require a Proxy Ingress/Egress Tunnel Router (PxTR) LISP infrastructure for access to non-LISP IPv6 addresses. (The PxTR infrastructure can still be configured as a resiliency mechanism if desired.)

The LISP site registers to two map resolver/map server (MR/MS) devices in the network core using both IPv4 and IPv6 locators. The topology used in this multihomed LISP configuration is shown in the figure below.

Figure 8: Multihomed LISP Site with Two xTRs, Each with an IPv4 and an IPv6 RLOC and each with an IPv4 and an IPv6 EID



The components illustrated in the topology shown in the figure are described below:

- **LISP site:**

- Two CPE routers make up the LISP site: xTR-1 and xTR-2.
- Both CPE routers function as LISP xTRs (that is, an ITR and an ETR).
- The LISP site is authoritative for both the IPv4 EID prefix of 172.16.1.0/24 and the IPv6 EID prefix 2001:db8:a::/48.
- Each LISP xTR has a single IPv4 RLOC connection and a single IPv6 RLOC connection to the core: the RLOC connections for xTR-1 to SP1 include an IPv4 RLOC, 10.1.1.2/30, and an IPv6

RLOC, 2001:db8:e000:1::2/64. The xTR-2 connections to SP2 include IPv4 RLOC 10.2.1.2/30 and IPv6 RLOC 2001:db8:f000:1::2/64.

- For this multihomed case, the LISP site policy specifies equal load-sharing between service provider (SP) links for ingress traffic engineering.

- **Mapping system:**

- Two map resolver/map server systems are assumed to be available for the LISP xTR to configure. The MR/MSs have IPv4 RLOCs 10.10.10.10 and 10.10.30.10 and IPv6 RLOCs 2001:db8:e000:2::1 and 2001:db8:f000:2::1.
- Mapping services are assumed to be provided as part of this LISP solution via a private mapping system or as a public LISP mapping system. From the perspective of the configuration of these LISP site xTRs, there is no difference.



Note Map resolver and map server configurations are not shown here. See the "Configure a Private LISP Mapping System Using a Standalone Map Resolver/Map Server" section for information about map resolver and map server configuration.

- PxTR services are not required in this example since both xTRs have dual-stack connectivity to the core.

Perform the steps in this task (once through for each xTR in the LISP site) to enable and configure LISP ITR and ETR (xTR) functionality when using a LISP map resolver and map server for mapping services. The example configurations at the end of this task show the full configuration for two xTRs (xTR1 and xTR2).

SUMMARY STEPS

1. **configure terminal**
2. **router lisp**
3. **database-mapping** *EID-prefix/prefix-length locator priority priority weight weight*
4. Repeat Step 3 to configure a second IPv4 RLOC for the same xTR and IPv4 EID prefix.
5. Repeat Step 3 and Step 4 to configure the same RLOC connections, again, for the same xTR but, when repeating these two steps, associate the IPv6 EID prefix, 2001:db8:a::/48, instead of the IPv4 EID prefix.
6. Repeat Step 3, Step 4, and Step 5 to configure the second set of IPv4 and IPv6 RLOC connections on the same xTR for both the IPv4 and IPv6 EID prefixes.
7. **ipv4 itr**
8. **ipv4 etr**
9. **ipv4 itr map-resolver** *map-resolver-address*
10. Repeat Step 9 to configure a second locator address of the LISP map resolver.
11. Repeat Step 9 and Step 10 to configure the IPv6 locator addresses of the LISP two map resolvers.
12. **ipv4 etr map-server** *map-server-address key key-type authentication-key*
13. Repeat Step 12 to configure a second locator address of the map server.
14. Repeat Step 12 and Step 13 to configure the IPv6 locator addresses of the two map servers.
15. **ipv6 itr**
16. **ipv6 etr**
17. **ipv6 itr map-resolver** *map-resolver-address*

18. Repeat Step 17 to configure a second IPv6 locator address of the LISP map resolver.
19. Repeat Step 17 and Step 18 to configure the IPv6 (instead of IPv4) locator addresses for the two map resolvers to which this router will send Map-Request messages for IPv6 EID-to-RLOC mapping resolutions.
20. **ipv6 etr map-server** *map-server-address* **key** *key-type authentication-key*
21. Repeat Step 20 to configure a second locator address of the LISP map server.
22. Repeat Steps 20 and 21 to configure the IPv6 locator addresses of the two map servers for which this router, acting as an IPv6 LISP ETR, will use to register to the LISP mapping system.
23. **exit**
24. **ip route** *ipv4-prefix next-hop*
25. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 2	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 3	database-mapping <i>EID-prefix/prefix-length locator</i> priority <i>priority</i> weight <i>weight</i> Example: <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50</pre>	Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site. <ul style="list-style-type: none"> • In this example, a single IPv4 EID prefix, 172.16.1.0/24, and a single IPv6 prefix, 2001:db8:a::/48, are being associated with a LISP site that contains two separate xTRs that each have a single IPv4 RLOC connection and a single IPv6 connection to the core. In this first database-mapping step example, xTR-1 is configured with an IPv4 RLOC connection to SP1 (10.1.1.2) and an IPv6 RLOC connection to SP1 (2001:db8:e000:1::2/64.) while xTR-2 has an IPv4 RLOC connection of 10.2.1.2 to SP2 and an IPv6 RLOC connection of 2001:db8:f000:1::2/64 to SP2. The <i>weight</i> argument of 50 signifies that a second database-mapping command is to be configured in the next step.

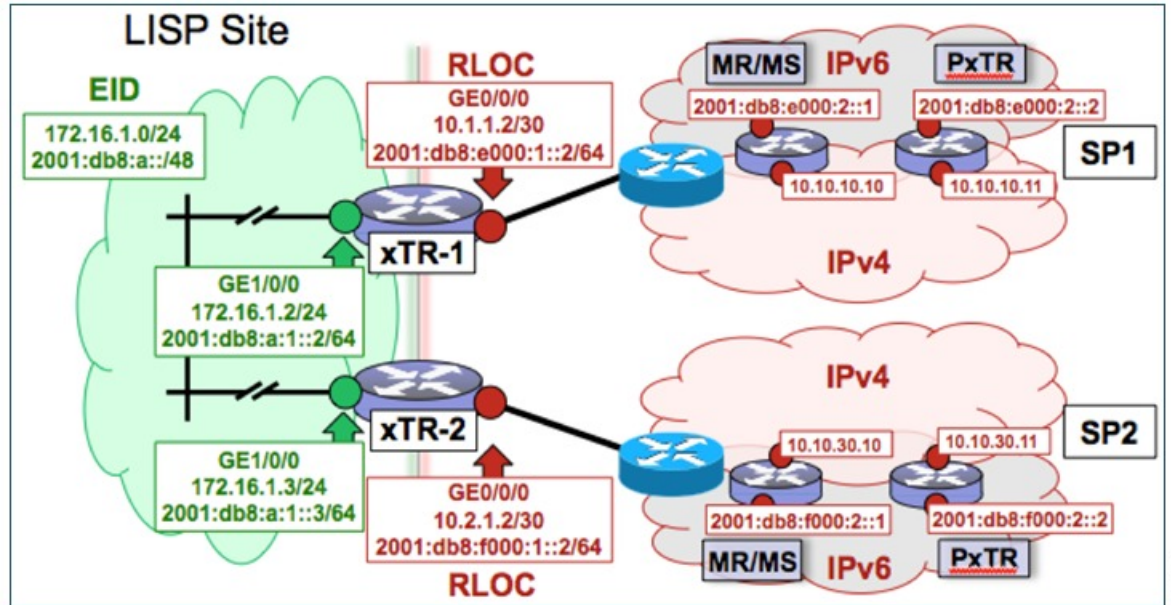
	Command or Action	Purpose
		<p>Note Eight database-mapping commands are required for each xTR to indicate to the mapping system that both the IPv4 and IPv6 EID prefixes are reachable at this LISP site via both the two IPv4 RLOCs and the two IPv6 RLOCs. In this example, one IPv4 RLOC and one IPv6 RLOC are local (connected) to one xTR and the others are local (connected) to the other xTR.</p>
Step 4	<p>Repeat Step 3 to configure a second IPv4 RLOC for the same xTR and IPv4 EID prefix.</p> <p>Example:</p> <pre>Router(config-router-lisp)# database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50</pre>	<p>Configures an EID-to-RLOC mapping relationship and its associated traffic policy for an xTR on this LISP site.</p> <ul style="list-style-type: none"> In this step example, the second RLOC connection for xTR-1 has an IPv4 RLOC connection to SP2 (10.2.1.2). <p>Note When a LISP site contains multiple xTRs, all xTRs must be configured with identical database-mapping commands to provide the mapping system with consistent information about EID-to-RLOC mappings.</p>
Step 5	<p>Repeat Step 3 and Step 4 to configure the same RLOC connections, again, for the same xTR but, when repeating these two steps, associate the IPv6 EID prefix, 2001:db8:a::/48, instead of the IPv4 EID prefix.</p> <p>Example:</p> <pre>Router(config-router-lisp)# database-mapping 2001:db8:a::/48 10.1.1.2 priority 1 weight 50</pre> <p>Example:</p> <pre>Router(config-router-lisp)# database-mapping 2001:db8:a::/48 10.2.1.2 priority 1 weight 50</pre>	—
Step 6	<p>Repeat Step 3, Step 4, and Step 5 to configure the second set of IPv4 and IPv6 RLOC connections on the same xTR for both the IPv4 and IPv6 EID prefixes.</p>	—
Step 7	<p>ipv4 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr</pre>	<p>Enables LISP ITR functionality for the IPv4 address family.</p>
Step 8	<p>ipv4 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr</pre>	<p>Enables LISP ETR functionality for the IPv4 address family.</p>

	Command or Action	Purpose
Step 9	<p>ipv4 itr map-resolver <i>map-resolver-address</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 10.10.10.10</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has both IPv4 and IPv6 RLOC connectivity, the map resolver is reachable via both IPv4 and IPv6 locator addresses. (See the <i>LISP Command Reference</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 10	<p>Repeat Step 9 to configure a second locator address of the LISP map resolver.</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 10.10.30.10</pre>	<p>Configures a second locator address for the LISP map resolver to which this router will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions.</p>
Step 11	<p>Repeat Step 9 and Step 10 to configure the IPv6 locator addresses of the LISP two map resolvers.</p>	—
Step 12	<p>ipv4 etr map-server <i>map-server-address</i> key <i>key-type</i> <i>authentication-key</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr map-server 10.10.10.10 key 0 some-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key that this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> In this example, a second xTR can be registered to the same two map servers using the same authentication key. The map server must be configured with EID prefixes matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has both IPv4 and IPv6 RLOC connectivity, the map server is reachable via both IPv4 and IPv6 locator addresses. (See the <i>LISP Command Reference</i> for more details.)</p> <p>Note Up to two map servers may be configured if multiple map servers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 13	<p>Repeat Step 12 to configure a second locator address of the map server.</p>	<p>Configures a second IPv4 locator address of the LISP map server and the authentication key that this router, acting</p>

	Command or Action	Purpose
	<p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr map-server 10.10.30.10 key 0 some-key</pre>	as an IPv4 LISP ETR, will use to register with the LISP mapping system.
Step 14	<p>Repeat Step 12 and Step 13 to configure the IPv6 locator addresses of the two map servers.</p> <p>Example:</p> <pre>ipv4 etr map-server 2001:db8:e000:2::1 key 0 some-xtr-key</pre> <p>Example:</p> <pre>ipv4 etr map-server 2001:db8:f000:2::1 key 0 some-xtr-key</pre>	—
Step 15	<p>ipv6 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr</pre>	Enables LISP ITR functionality for the IPv6 address family.
Step 16	<p>ipv6 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 etr</pre>	Enables LISP ETR functionality for the IPv6 address family.
Step 17	<p>ipv6 itr map-resolver map-resolver-address</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr map-resolver 10.10.10.10</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send Map-Request messages for IPv6 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has both IPv4 and IPv6 RLOC connectivity, the map resolver is reachable via both IPv4 and IPv6 locator addresses. (See the <i>LISP Command Reference</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 18	<p>Repeat Step 17 to configure a second IPv6 locator address of the LISP map resolver.</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr map-resolver 10.10.30.10</pre>	Configures a second locator address of the map resolver to which this router will send Map-Request messages for IPv6 EID-to-RLOC mapping resolutions.
Step 19	<p>Repeat Step 17 and Step 18 to configure the IPv6 (instead of IPv4) locator addresses for the two map resolvers to</p>	—

	Command or Action	Purpose
	<p>which this router will send Map-Request messages for IPv6 EID-to-RLOC mapping resolutions.</p> <p>Example:</p> <pre>ipv6 itr map-resolver 2001:db8:e000:2::1</pre> <p>Example:</p> <pre>ipv6 itr map-resolver 2001:db8:f000:2::1</pre>	
Step 20	<p>ipv6 etr map-server map-server-address key key-type authentication-key</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 etr map-server 10.10.10.10 key 0 some-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key that this router, acting as an IPv6 LISP ETR, will use to register to the LISP mapping system.</p> <ul style="list-style-type: none"> • In this example, a second xTR can be registered to the same two map servers using the same authentication key. • The map server must be configured with EID prefixes matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has both IPv4 and IPv6 RLOC connectivity, the map server is reachable via both IPv4 and IPv6 locator addresses. (See the <i>LISP Command Reference</i> for more details.)</p> <p>Note Up to two map servers may be configured if multiple map servers are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 21	<p>Repeat Step 20 to configure a second locator address of the LISP map server.</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 etr map-server 10.10.30.10 key 0 some-key</pre>	<p>Configures a second locator address for the LISP map server and an authentication key that this router, acting as an IPv6 LISP ETR, will use to register with the LISP mapping system.</p>
Step 22	<p>Repeat Steps 20 and 21 to configure the IPv6 locator addresses of the two map servers for which this router, acting as an IPv6 LISP ETR, will use to register to the LISP mapping system.</p> <p>Example:</p> <pre>ipv6 etr map-server 2001:db8:e000:2::1 key 0 some-xtr-key</pre> <p>Example:</p> <pre>ipv6 etr map-server 2001:db8:f000:2::1 key 0 some-xtr-key</pre>	—

	Command or Action	Purpose
Step 23	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 24	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> • All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> • LISP-encapsulated to a LISP site when traffic is LISP-to-LISP • natively forwarded when traffic is LISP-to-non-LISP • Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP EID and the destination matches one of the following entries: <ul style="list-style-type: none"> • a current map-cache entry • a default route with a legitimate next-hop • no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
Step 25	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	Exits global configuration mode.

Example:**Figure 9: Multihomed LISP Site with Two xTRs, Each with an IPv4 and an IPv6 RLOC and each with an IPv4 and an IPv6 EID**

The examples below show the complete configuration for the LISP topology illustrated in the figure above and in this task:

Example configuration for xTR-1:

```

!
hostname xTR-1
!
no ip domain lookup
ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 ip address 172.17.1.1 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
 description Link to SP1 (RLOC)
 ip address 10.1.1.2 255.255.255.252
 ipv6 address 2001:db8:e000:1::2/64
!
interface GigabitEthernet1/0/0
 description Link to Site (EID)
 ip address 172.16.1.2 255.255.255.0
 ipv6 address 2001:db8:a:1::2/64
!
router lisp
 database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
 database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
 database-mapping 2001:db8:a::/48 10.1.1.2 priority 1 weight 50

```

```

database-mapping 2001:db8:a::/48 10.2.1.2 priority 1 weight 50
database-mapping 172.16.1.0/24 2001:db8:e000:1::2 priority 1 weight 50
database-mapping 172.16.1.0/24 2001:db8:f000:1::2 priority 1 weight 50
database-mapping 2001:db8:a::/48 2001:db8:e000:1::2 priority 1 weight 50
database-mapping 2001:db8:a::/48 2001:db8:f000:1::2 priority 1 weight 50
ipv4 itr
ipv4 etr
ipv4 itr map-resolver 10.10.10.10
ipv4 itr map-resolver 10.10.30.10
ipv4 itr map-resolver 2001:db8:e000:2::1
ipv4 itr map-resolver 2001:db8:f000:2::1
ipv4 etr map-server 10.10.10.10 key 0 some-xtr-key
ipv4 etr map-server 10.10.30.10 key 0 some-xtr-key
ipv4 etr map-server 2001:db8:e000:2::1 key 0 some-xtr-key
ipv4 etr map-server 2001:db8:f000:2::1 key 0 some-xtr-key
ipv6 itr
ipv6 etr
ipv6 itr map-resolver 10.10.10.10
ipv6 itr map-resolver 10.10.30.10
ipv6 itr map-resolver 2001:db8:e000:2::1
ipv6 itr map-resolver 2001:db8:f000:2::1
ipv6 etr map-server 10.10.10.10 key 0 some-xtr-key
ipv6 etr map-server 10.10.30.10 key 0 some-xtr-key
ipv6 etr map-server 2001:db8:e000:2::1 key 0 some-xtr-key
ipv6 etr map-server 2001:db8:f000:2::1 key 0 some-xtr-key
exit
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
!
ipv6 route ::/0 2001:db8:e000:1::1
!

```

Example configuration for xTR-2:

```

!
hostname xTR-2
!
no ip domain lookup
ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 ip address 172.17.1.2 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
 description Link to SP2 (RLOC)
 ip address 10.2.1.2 255.255.255.252
 ipv6 address 2001:db8:f000:1::2/64
!
interface GigabitEthernet1/0/0
 description Link to Site (EID)
 ip address 172.16.1.3 255.255.255.0
 ipv6 address 2001:db8:a:1::3/64
!
router lisp
database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
database-mapping 2001:db8:a::/48 10.1.1.2 priority 1 weight 50
database-mapping 2001:db8:a::/48 10.2.1.2 priority 1 weight 50
database-mapping 172.16.1.0/24 2001:db8:e000:1::2 priority 1 weight 50
database-mapping 172.16.1.0/24 2001:db8:f000:1::2 priority 1 weight 50

```

```

database-mapping 2001:db8:a::/48 2001:db8:e000:1::2 priority 1 weight 50
database-mapping 2001:db8:a::/48 2001:db8:f000:1::2 priority 1 weight 50
ipv4 itr
ipv4 etr
ipv4 itr map-resolver 10.10.10.10
ipv4 itr map-resolver 10.10.30.10
ipv4 itr map-resolver 2001:db8:e000:2::1
ipv4 itr map-resolver 2001:db8:f000:2::1
ipv4 etr map-server 10.10.10.10 key 0 some-xtr-key
ipv4 etr map-server 10.10.30.10 key 0 some-xtr-key
ipv4 etr map-server 2001:db8:e000:2::1 key 0 some-xtr-key
ipv4 etr map-server 2001:db8:f000:2::1 key 0 some-xtr-key
ipv6 itr
ipv6 etr
ipv6 itr map-resolver 10.10.10.10
ipv6 itr map-resolver 10.10.30.10
ipv6 itr map-resolver 2001:db8:e000:2::1
ipv6 itr map-resolver 2001:db8:f000:2::1
ipv6 etr map-server 10.10.10.10 key 0 some-xtr-key
ipv6 etr map-server 10.10.30.10 key 0 some-xtr-key
ipv6 etr map-server 2001:db8:e000:2::1 key 0 some-xtr-key
ipv6 etr map-server 2001:db8:f000:2::1 key 0 some-xtr-key
exit
!
ip route 0.0.0.0 0.0.0.0 10.2.1.1
!
ipv6 route ::/0 2001:db8:f000:1::1
!

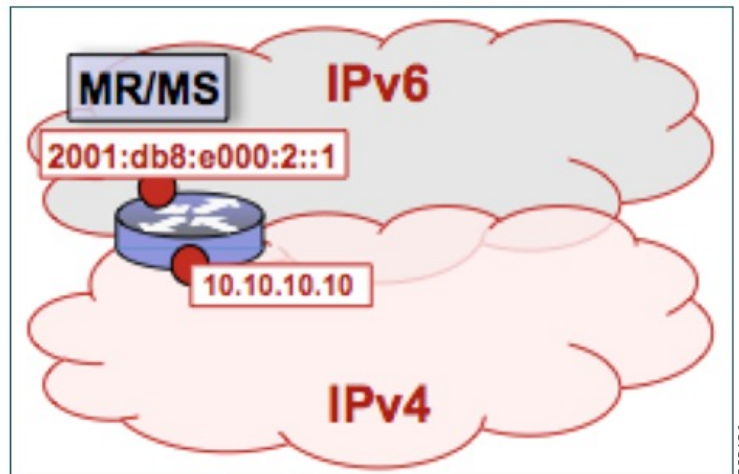
```

Configure a Private LISP Mapping System Using a Standalone Map Resolver/Map Server

Perform this task to configure and enable standalone LISP map resolver/map server (MR/MS) functionality for both IPv4 and IPv6 address families. In this task, a Cisco device is configured as a standalone MR/MS for a private LISP mapping system. Because the MR/MS is configured as a standalone device, it has no need for LISP alternative logical topology (ALT) connectivity. All relevant LISP sites must be configured to register with this map server so that this map server has full knowledge of all registered EID prefixes within the (assumed) private LISP system. However, because this device is functioning as a map resolver/map server, the data structure associated with an ALT virtual routing and forwarding (VRF) table must still be configured to hold LISP EIDs for registered sites.

The map resolver/map server is configured with both IPv4 and IPv6 RLOC addresses. The topology used in this most basic LISP MR/MS configuration is shown in the figure below.

Figure 10: Standalone LISP Map Resolver/Map Server with both IPv4 and IPv6 RLOCs



The components illustrated in the topology shown in the figure are described below, although the map resolver is configured separately:

Mapping System

- The LISP device is configured to function as a standalone map resolver/map server (MR/MS).
- The xTRs in the LISP site are assumed to be registered to this map server. That is, the xTR registers the IPv4 EID prefix of 172.16.1.0/24 and, when IPv6 EIDs are used, the xTR also registers the IPv6 EID of prefix 2001:db8:a::/48.
- The MR/MS has an IPv4 locator of 10.10.10.10/24 and an IPv6 locator of 2001:db8:e000:2::1/64.

SUMMARY STEPS

1. **configure terminal**
2. **vrf definition** *vrf-name*
3. **address-family ipv4** [**unicast**]
4. **exit-address-family**
5. **address-family ipv6**
6. **exit-address-family**
7. **exit**
8. **router lisp**
9. **ipv4 alt-vrf** *vrf-name*
10. **ipv4 map-server**
11. **ipv4 map-resolver**
12. **ipv6 alt-vrf** *vrf-name*
13. **ipv6 map-server**
14. **ipv6 map-resolver**
15. **site** *site-name*
16. **eid-prefix** *EID-prefix*
17. **authentication-key** [*key-type*] *authentication-key*
18. **exit**

19. Repeat Steps 15 through 18 to configure additional LISP sites.
20. `exit`
21. `ip route ipv4-prefix next-hop`
22. `ipv6 route ipv6-prefix next-hop`
23. `exit`

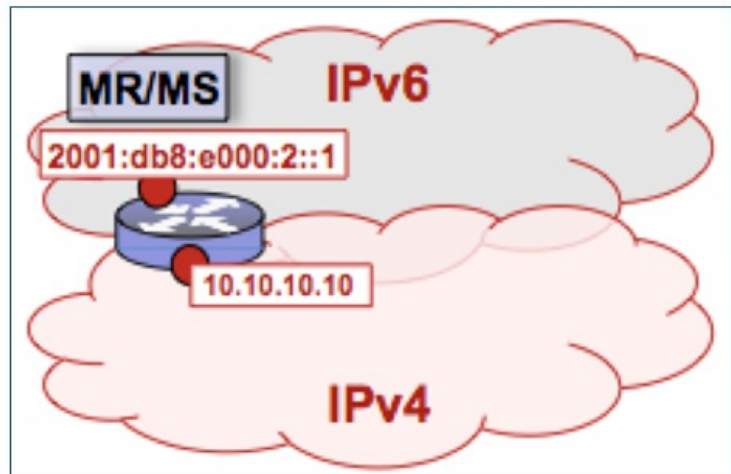
DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 2	vrf definition vrf-name Example: <pre>Router(config)# vrf definition lisp</pre>	Creates a virtual routing and forwarding (VRF) table and enters VRF configuration mode. <ul style="list-style-type: none"> • Use the <i>vrf-name</i> argument to specify a name to be assigned to the VRF table. In this example, a VRF table named <i>lisp</i> is created to hold EID prefixes.
Step 3	address-family ipv4 [unicast] Example: <pre>Router(config-vrf)# address-family ipv4</pre>	Enters VRF IPv4 address family configuration mode to specify an IPv4 address family for a VRF table. <ul style="list-style-type: none"> • In this example, the VRF table named <i>lisp</i> handles IPv4 EID prefixes.
Step 4	exit-address-family Example: <pre>Router(config-vrf-af)# exit-address-family</pre>	Exits VRF IPv4 address family configuration mode and returns to VRF configuration mode.
Step 5	address-family ipv6 Example: <pre>Router(config-vrf)# address-family ipv6</pre>	Enters VRF IPv6 address family configuration mode to specify an IPv6 address family for a VRF table. <ul style="list-style-type: none"> • In this example, the VRF table named <i>lisp</i> handles IPv6 EID prefixes.
Step 6	exit-address-family Example: <pre>Router(config-vrf-af)# exit-address-family</pre>	Exits VRF IPv6 address family configuration mode and returns to VRF configuration mode.
Step 7	exit Example: <pre>Router(config-vrf)# exit</pre>	Exits VRF configuration mode and enters global configuration mode.

	Command or Action	Purpose
Step 8	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 9	ipv4 alt-vrf vrf-name Example: <pre>Router(config-router-lisp)# ipv4 alt-vrf lisp</pre>	Associates a VRF table with the LISP ALT for IPv4 EIDs. <ul style="list-style-type: none"> In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 10	ipv4 map-server Example: <pre>Router(config-router-lisp)# ipv4 map-server</pre>	Enables LISP map server functionality for EIDs in the IPv4 address family.
Step 11	ipv4 map-resolver Example: <pre>Router(config-router-lisp)# ipv4 map-resolver</pre>	Enables LISP map resolver functionality for EIDs in the IPv4 address family.
Step 12	ipv6 alt-vrf vrf-name Example: <pre>Router(config-router-lisp)# ipv6 alt-vrf lisp</pre>	Associates a VRF table with the LISP ALT for IPv6 EIDs. <ul style="list-style-type: none"> In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 13	ipv6 map-server Example: <pre>Router(config-router-lisp)# ipv6 map-server</pre>	Enables LISP map server functionality for EIDs in the IPv6 address family.
Step 14	ipv6 map-resolver Example: <pre>Router(config-router-lisp)# ipv6 map-resolver</pre>	Enables LISP map resolver functionality for EIDs in the IPv6 address family.
Step 15	site site-name Example: <pre>Router(config-router-lisp)# site Site-1</pre>	Specifies a LISP site named Site-1 and enters LISP site configuration mode. <p>Note A LISP site name is locally significant to the map server on which it is configured. It has no relevance anywhere else. This name is used solely as an administrative means of associating one or more EID prefixes with an authentication key and other site-related mechanisms.</p>
Step 16	eid-prefix EID-prefix Example:	Configures an IPv4 or IPv6 EID prefix associated with this LISP site.

	Command or Action	Purpose
	<pre>Router(config-router-lisp-site)# eid-prefix 172.16.1.0/24</pre>	<ul style="list-style-type: none"> Repeat this step as necessary to configure additional EID prefixes under this LISP sites. In this step example, only an IPv4 EID prefix is configured but to complete the configuration, an IPv6 EID prefix must also be configured. <p>Note The LISP ETR must be configured with matching EID prefixes and an identical authentication key.</p> <p>Note Additional eid-prefix command configuration options are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 17	<p>authentication-key [<i>key-type</i>] <i>authentication-key</i></p> <p>Example:</p> <pre>Router(config-router-lisp-site)# authentication-key 0 some-key</pre>	<p>Configures the authentication key associated with this site.</p> <p>Note The LISP ETR must be configured with matching EID prefixes and an identical authentication key.</p> <p>Note The authentication-key can be configured with Type 6 encryption. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 18	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp-site)# exit</pre>	<p>Exits LISP site configuration mode and returns to LISP configuration mode.</p>
Step 19	<p>Repeat Steps 15 through 18 to configure additional LISP sites.</p>	—
Step 20	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	<p>Exits LISP configuration mode and returns to global configuration mode.</p>
Step 21	<p>ip route <i>ipv4-prefix</i> <i>next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1</pre>	<p>Configures an IPv4 static route.</p> <ul style="list-style-type: none"> In this example, a default route to the upstream next hop for all IPv4 destinations is created.
Step 22	<p>ipv6 route <i>ipv6-prefix</i> <i>next-hop</i></p> <p>Example:</p> <pre>Router(config)# ipv6 route ::/0 2001:db8:e000:1::1</pre>	<p>Configures an IPv6 static route.</p> <ul style="list-style-type: none"> In this example, a default route to the upstream next hop for all IPv6 destinations is created.

	Command or Action	Purpose
Step 23	exit Example: Router(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Example:*Figure 11: Standalone LISP Map Resolver/Map Server with both IPv4 and IPv6 RLOCs*

The example below shows the complete configuration for the LISP topology illustrated in the figure above and in this task. However, this example is for a full configuration of a standalone LISP MR/MS and includes some basic IPv4 and IPv6 configuration not covered in this task:

```

!
hostname MR-MS
!
vrf definition lisp
!
  address-family ipv4
  exit-address-family
  !
  address-family ipv6
  exit-address-family
!
no ip domain lookup
ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
  ip address 172.17.2.1 255.255.255.255
!
interface LISP0
!
interface GigabitEthernet0/0/0
  description Link to SP1 (RLOC)
  ip address 10.10.10.10 255.255.255.0

```

```

    ipv6 address 2001:db8:e000:2::1/64
    !
router lisp
  site Site-1
    authentication-key some-key
    eid-prefix 172.16.1.0/24
    eid-prefix 2001:db8:a::/48
    exit
  !
  site Site-2
    authentication-key another-key
    eid-prefix 172.16.2.0/24
    eid-prefix 2001:db8:b::/48
    exit
  !
!---more LISP site configs---
!
  ipv4 map-server
  ipv4 map-resolver
  ipv4 alt-vrf lisp
  ipv6 map-server
  ipv6 map-resolver
  ipv6 alt-vrf lisp
  exit
!
ip route 0.0.0.0 0.0.0.0 10.10.10.1
!
ipv6 route ::/0 2001:db8:e000:2::fof

```

Configure a Public Mapping System Using Separate ALT-Connected Map Resolver and Map Server Devices

The following tasks show how to configure a map resolver (MR) and a map server (MS) on separate devices, each using LISP alternative logical topology (ALT) connectivity. The MR and MS share their EID prefix information via the LISP ALT connectivity, which is typical of a public LISP deployment model where higher performance and scalability (for tasks such as the handling of Map-Request messages) is required. The LISP ALT is implemented as an overlay virtualized network using GRE tunnels and BGP, which allows for separation of EID prefixes from the underlying core network.

Configuring an ALT-Connected LISP Map Resolver

Before you begin

Perform this task to configure LISP alternative logical topology (ALT) map resolver functionality for both IPv4 and IPv6 address family mapping services on Cisco IOS XE Everest 16.6.1 and later releases.

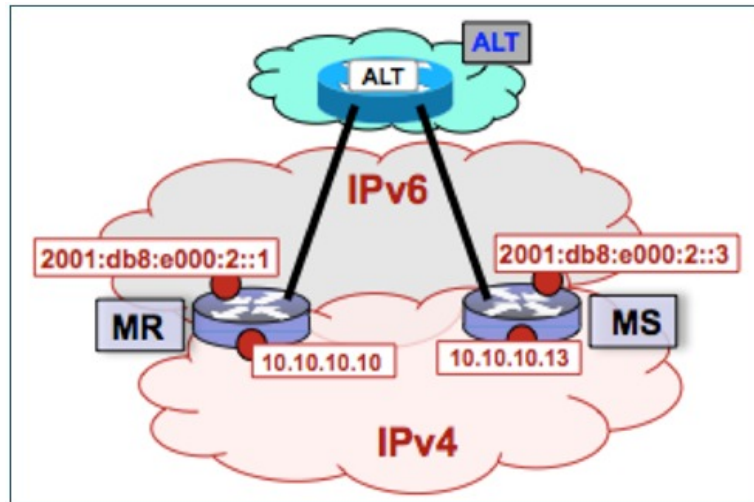


Note

You must also configure an ALT-connected LISP map server (see the Configuring an ALT-Connected LISP Map Server task).

In the figure below, the map resolver (MR) and map server (MS) are configured on separate devices and share their EID prefix information via connectivity.

Figure 12: ALT-Connected LISP Map Resolver and Map Server, each having both an IPv4 and an IPv6 RLOC



The map resolver illustrated in the topology shown in the figure is described below; the map server and LISP ALT are configured in separate tasks:

Mapping System

- Two LISP devices are configured, one as an MS and the other as an MR.
- The MS has an IPv4 locator of 10.10.10.13/24 and an IPv6 locator of 2001:db8:e000:2::3/64.
- The MR has an IPv4 locator of 10.10.10.10/24 and an IPv6 locator of 2001:db8:e000:2::1/64.
- Assume that the xTRs in the LISP site register to this map server. That is, the xTR registers the IPv4 EID-prefix of 172.16.1.0/24 and, when IPv6 EIDs are used, the xTR registers the IPv6 EID-prefix of 2001:db8:a::/48.



Note The configuration of the xTR must be changed to use the MS RLOC for its map server configuration and the MR RLOC for its map resolver configuration. For example:

- `ipv4 itr map-resolver 10.10.10.10`
- `ipv4 etr map-server 10.10.10.13 key 0 some-key`

Other Infrastructure

- The MR has IPv4 and IPv6 tunnel endpoints in the VRF table (named lisp) of 192.168.1.1/30 and 2001:db8:fff::1/64, respectively, and the MS has IPv4 and IPv6 tunnel endpoints of 192.168.1.2/30 and 2001:db8:fff::2/64, respectively, in the same VRF table. This tunnel is used for the ALT.

SUMMARY STEPS

1. **configure terminal**
2. **vrf definition** *vrf-name*
3. **rd** *route-distinguisher*
4. **address-family ipv4** [**unicast**]

5. **exit-address-family**
6. **address-family ipv6**
7. **exit-address-family**
8. **exit**
9. **interface** *type number*
10. **vrf forwarding** *vrf-name*
11. **ip address** *ip-address mask*
12. **ipv6 address** *ipv6-address/mask*
13. **tunnel source** *interface-type interface-number*
14. **tunnel destination** *ipv4-address*
15. **exit**
16. **router lisp**
17. **ipv4 map-resolver**
18. **ipv4 alt-vrf** *vrf-name*
19. **ipv6 map-resolver**
20. **ipv6 alt-vrf** *vrf-name*
21. **exit**
22. **router bgp** *autonomous-system-number*
23. **address-family ipv4** [**unicast** | **multicast** | **vrf** *vrf-name*]
24. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
25. **neighbor** *ip-address* **activate**
26. **exit**
27. **address-family ipv6** **vrf** *vrf-name*
28. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
29. **neighbor** *ip-address* **activate**
30. **exit**
31. **exit**
32. **ip route** *ipv4-prefix next-hop*
33. **ipv6 route** *ipv6-prefix next-hop*
34. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# <code>configure terminal</code>	Enters global configuration mode.
Step 2	vrf definition <i>vrf-name</i> Example: Router(config)# <code>vrf definition lisp</code>	Creates a virtual routing and forwarding (VRF) table and enters VRF configuration mode. <ul style="list-style-type: none"> • Use the <i>vrf-name</i> argument to specify a name to be assigned to the VRF. In this example, a VRF named <code>lisp</code> is created to hold EID prefixes.

	Command or Action	Purpose
Step 3	rd <i>route-distinguisher</i> Example: Router(config-vrf)# rd 1:1	Creates routing and forwarding tables for a VRF.
Step 4	address-family ipv4 [<i>unicast</i>] Example: Router(config-vrf)# address-family ipv4	Enters VRF IPv4 address family configuration mode to specify an IPv4 address family for a VRF table. <ul style="list-style-type: none"> In this example, the VRF table named lisp handles IPv4 EID prefixes.
Step 5	exit-address-family Example: Router(config-vrf-af)# exit-address-family	Exits VRF IPv4 address family configuration mode and returns to VRF configuration mode.
Step 6	address-family ipv6 Example: Router(config-vrf)# address-family ipv6	Enters VRF IPv6 address family configuration mode to specify an IPv6 address family for a VRF table. <ul style="list-style-type: none"> In this example, the VRF table named lisp handles IPv6 EID prefixes.
Step 7	exit-address-family Example: Router(config-vrf-af)# exit-address-family	Exits VRF IPv6 address family configuration mode and returns to VRF configuration mode.
Step 8	exit Example: Router(config-vrf)# exit	Exits VRF configuration mode and enters global configuration mode.
Step 9	interface <i>type number</i> Example: Router(config)# interface tunnel 192	Specifies the interface type of tunnel and the interface number and enters interface configuration mode.
Step 10	vrf forwarding <i>vrf-name</i> Example: Router(config-if)# vrf forwarding lisp	Associates a VRF instance configured in Step 2 with the tunnel interface configured in Step 9. <ul style="list-style-type: none"> When the interface is bound to a VRF, previously configured IP addresses are removed, and the interface is disabled.
Step 11	ip address <i>ip-address mask</i> Example: Router(config-if)# ip address 192.168.1.1 255.255.255.252	Configures an IPv4 address for the tunnel interface.
Step 12	ipv6 address <i>ipv6-address/mask</i> Example:	Configures an IPv6 address for the tunnel interface.

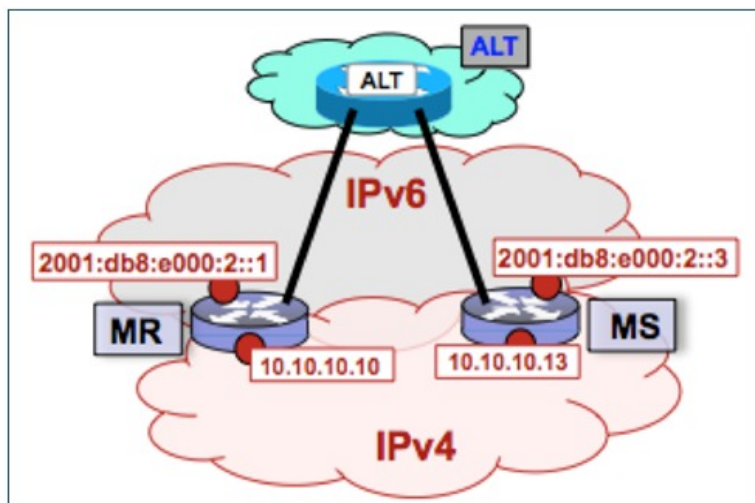
	Command or Action	Purpose
	Router(config-if)# ipv6 address 2001:db8:ffff::1/64	
Step 13	tunnel source <i>interface-type interface-number</i> Example: Router(config-if)# tunnel source GigabitEthernet 0/0/0	Configures the tunnel source.
Step 14	tunnel destination <i>ipv4-address</i> Example: Router(config-if)# tunnel destination 10.10.10.13	Configures the tunnel destination IPv4 address for the tunnel interface.
Step 15	exit Example: Router(config-if)# exit	Exits interface configuration mode and enters global configuration mode.
Step 16	router lisp Example: Router(config)# router lisp	Enters LISP configuration mode (software only).
Step 17	ipv4 map-resolver Example: Router(config-router-lisp)# ipv4 map-resolver	Enables LISP map resolver functionality for EIDs in the IPv4 address family.
Step 18	ipv4 alt-vrf <i>vrf-name</i> Example: Router(config-router-lisp)# ipv4 alt-vrf lisp	Associates a VRF table with the LISP ALT for IPv4 EIDs. <ul style="list-style-type: none"> • In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 19	ipv6 map-resolver Example: Router(config-router-lisp)# ipv6 map-resolver	Enables LISP map resolver functionality for EIDs in the IPv6 address family.
Step 20	ipv6 alt-vrf <i>vrf-name</i> Example: Router(config-router-lisp)# ipv6 alt-vrf lisp	Associates a VRF table with the LISP ALT for IPv6 EIDs. <ul style="list-style-type: none"> • In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 21	exit Example: Router(config-router-lisp)# exit	Exits LISP configuration mode and returns to global configuration mode.
Step 22	router bgp <i>autonomous-system-number</i> Example: Router(config)# router bgp 65010	Enters router configuration mode for the specified routing process.

	Command or Action	Purpose
Step 23	<p>address-family ipv4 [unicast multicast vrf vrf-name]</p> <p>Example:</p> <pre>Router(config-router)# address-family ipv4 vrf lisp</pre>	<p>Specifies the IPv4 address family and enters IPv4 address family configuration mode.</p> <ul style="list-style-type: none"> The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent commands. In this example, the VRF table named <i>lisp</i> (created in Step 2) is associated with the BGP IPv4 VRF that carries EID-prefixes in the LISP ALT.
Step 24	<p>neighbor ip-address remote-as autonomous-system-number</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 192.168.1.2 remote-as 65011</pre>	<p>Adds the IP address of the neighbor in the specified autonomous system to the IPv4 multiprotocol BGP neighbor table of the local router.</p>
Step 25	<p>neighbor ip-address activate</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 192.168.1.2 activate</pre>	<p>Enables the neighbor to exchange prefixes for the IPv4 unicast address family.</p>
Step 26	<p>exit</p> <p>Example:</p> <pre>Router(config-router-af)# exit</pre>	<p>Exits IPv4 address family configuration mode and returns to router configuration mode.</p>
Step 27	<p>address-family ipv6 vrf vrf-name</p> <p>Example:</p> <pre>Router(config-router)# address-family ipv6 vrf lisp</pre>	<p>Specifies the IPv6 address family and enters IPv6 address family configuration mode.</p> <ul style="list-style-type: none"> The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent commands. In this example, the VRF table named <i>lisp</i> (created in Step 2) is associated with the BGP IPv6 VRF that carries EID-prefixes in the LISP ALT.
Step 28	<p>neighbor ip-address remote-as autonomous-system-number</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 2001:db8:ffff::2 remote-as 65011</pre>	<p>Adds the IPv6 address of the neighbor in the specified autonomous system to the IPv6 multiprotocol BGP neighbor table of the local router.</p>
Step 29	<p>neighbor ip-address activate</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 2001:db8:ffff::2 activate</pre>	<p>Enables the neighbor to exchange prefixes for the IPv6 unicast address family.</p>

	Command or Action	Purpose
Step 30	exit Example: Router(config-router-af)# exit	Exits address family configuration mode and returns to router configuration mode.
Step 31	exit Example: Router(config-router)# exit	Exits router configuration mode and returns to global configuration mode.
Step 32	ip route ipv4-prefix next-hop Example: Router(config)# ip route 0.0.0.0 0.0.0.0 10.10.10.1	Configures an IPv4 static route. <ul style="list-style-type: none">In this example, a default route to the upstream next hop for all IPv4 destinations is created.
Step 33	ipv6 route ipv6-prefix next-hop Example: Router(config)# ipv6 route ::/0 2001:db8:e000:2::f0f	Configures an IPv6 static route. <ul style="list-style-type: none">In this example, a default route to the upstream next hop for all IPv6 destinations is created.
Step 34	exit Example: Router(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Examples

Figure 13: ALT-Connected LISP Map Resolver and Map Server, each having both an IPv4 and an IPv6 RLOC



The example below shows the full configuration for a LISP map resolver including some basic IP and IPv6 configuration not included in the task table for this task:

!


```

vrf definition lisp
 rd 1:1
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
 !
 no ip domain lookup
 ip cef
 ipv6 unicast-routing
 ipv6 cef
 !
 interface Loopback0
  no ip address
 !
 interface Tunnel192
  vrf forwarding lisp
  ip address 192.168.1.1 255.255.255.252
  ipv6 address 2001:db8:ffff::1/64
  tunnel source GigabitEthernet 0/0/0
  tunnel destination 10.10.10.13
 !
 interface GigabitEthernet 0/0/0
  description Link to SP1 (RLOC)
  ip address 10.10.10.10 255.255.255.0
  ipv6 address 2001:db8:e000:2::1/64
 !
 router lisp
  ipv4 map-resolver
  ipv4 alt-vrf lisp
  ipv6 map-resolver
  ipv6 alt-vrf lisp
  exit
 !
 router bgp 65010
  bgp asnotation dot
  bgp log-neighbor-changes
  !
  address-family ipv4 vrf lisp
   neighbor 192.168.1.2 remote-as 65011
   neighbor 192.168.1.2 activate
  exit-address-family
  !
  address-family ipv6 vrf lisp
   neighbor 2001:db8:ffff::2 remote-as 65011
   neighbor 2001:db8:ffff::2 activate
  exit-address-family
 !
 ip route 0.0.0.0 0.0.0.0 10.10.10.1
 !
 ipv6 route ::/0 2001:db8:e000:2::f0f
 !

```

Configuring an ALT-Connected LISP Map Server

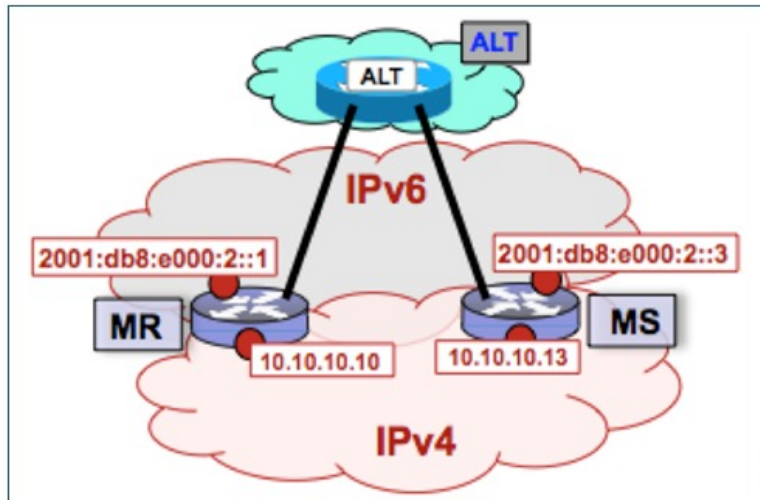
Perform this task to configure LISP alternative logical topology (ALT) map server functionality for both IPv4 and IPv6 address family mapping services.



Note You must also configure an ALT-connected LISP map resolver (see the Configuring an ALT-Connected LISP Map Resolver task).

In the figure below, the map resolver (MR) and map server (MS) are configured on separate devices and share their EID prefix information via connectivity.

Figure 14: ALT-Connected LISP Map Resolver and Map Server, each having both an IPv4 and an IPv6 RLOC



The map server illustrated in the topology shown in the figure is described below; the map resolver and LISP ALT are configured in separate tasks:

Mapping System

- Two LISP devices are configured, one as an MS and the other as an MR.
- The MS has an IPv4 locator of 10.10.10.13/24 and an IPv6 locator of 2001:db8:e000:2::3/64.
- The MR has an IPv4 locator of 10.10.10.10/24 and an IPv6 locator of 2001:db8:e000:2::1/64.
- Assume that the xTRs in the LISP site register to this map server. That is, the xTR registers the IPv4 EID-prefix of 172.16.1.0/24 and, when IPv6 EIDs are used, the xTR registers the IPv6 EID-prefix of 2001:db8:a::/48.



Note The configuration of the xTR must be changed to use the MS RLOC for its map server configuration and the MR RLOC for its map resolver configuration. For example:

- `ipv4 itr map-resolver 10.10.10.10`
- `ipv4 etr map-server 10.10.10.13 key 0 some-key`

Other Infrastructure

- The MR has IPv4 and IPv6 tunnel endpoints in the VRF table (named lisp) of 192.168.1.1/30 and 2001:db8:ffff::1/64, respectively, and the MS has IPv4 and IPv6 tunnel endpoints of 192.168.1.2/30 and 2001:db8:ffff::2/64, respectively, in the same VRF table. This tunnel is used for the ALT.

SUMMARY STEPS

1. **configure terminal**
2. **vrf definition** *vrf-name*
3. **rd** *route-distinguisher*
4. **address-family ipv4** [**unicast**]
5. **exit-address-family**
6. **address-family ipv6**
7. **exit-address-family**
8. **exit**
9. **interface** *type number*
10. **vrf forwarding** *vrf-name*
11. **ip address** *ip-address mask*
12. **ipv6 address** *ipv6-address/mask*
13. **tunnel source** *interface-type interface-number*
14. **tunnel destination** *ipv4-address*
15. **exit**
16. **router lisp**
17. **ipv4 map-server**
18. **ipv4 alt-vrf** *vrf-name*
19. **ipv6 map-server**
20. **ipv6 alt-vrf** *vrf-name*
21. **site** *site-name*
22. **eid-prefix** *EID-prefix*
23. **authentication-key** *key-type authentication-key*
24. **exit**
25. Repeat Steps 21 through 24 to configure additional LISP sites.
26. **exit**
27. **router bgp** *autonomous-system-number*
28. **address-family ipv4** [**unicast** | **multicast** | **vrf** *vrf-name*]
29. **redistribute lisp**
30. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
31. **neighbor** *ip-address* **activate**
32. **exit**
33. **address-family ipv6** **vrf** *vrf-name*
34. **redistribute lisp**
35. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
36. **neighbor** *ip-address* **activate**
37. **exit**
38. **exit**
39. **ip route** *ipv4-prefix next-hop*

40. `ipv6 route ipv6-prefix next-hop`
 41. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# <code>configure terminal</code>	Enters global configuration mode.
Step 2	vrf definition vrf-name Example: Router(config)# <code>vrf definition lisp</code>	Creates a virtual routing and forwarding (VRF) table and enters VRF configuration mode. <ul style="list-style-type: none"> Use the <i>vrf-name</i> argument to specify a name to be assigned to the VRF. In this example, a VRF named <i>lisp</i> is created to hold EID prefixes.
Step 3	rd route-distinguisher Example: Router(config-vrf)# <code>rd 1:1</code>	Creates routing and forwarding tables for a VRF.
Step 4	address-family ipv4 [unicast] Example: Router(config-vrf)# <code>address-family ipv4</code>	Enters VRF IPv4 address family configuration mode to specify an IPv4 address family for a VRF table. <ul style="list-style-type: none"> In this example, the VRF table named <i>lisp</i> handles IPv4 EID prefixes.
Step 5	exit-address-family Example: Router(config-vrf-af)# <code>exit-address-family</code>	Exits VRF IPv4 address family configuration mode and returns to VRF configuration mode.
Step 6	address-family ipv6 Example: Router(config-vrf)# <code>address-family ipv6</code>	Enters VRF IPv6 address family configuration mode to specify an IPv6 address family for a VRF table. <ul style="list-style-type: none"> In this example, the VRF table named <i>lisp</i> handles IPv6 EID prefixes.
Step 7	exit-address-family Example: Router(config-vrf-af)# <code>exit-address-family</code>	Exits VRF IPv6 address family configuration mode and returns to VRF configuration mode.
Step 8	exit Example: Router(config-vrf)# <code>exit</code>	Exits VRF configuration mode and enters global configuration mode.

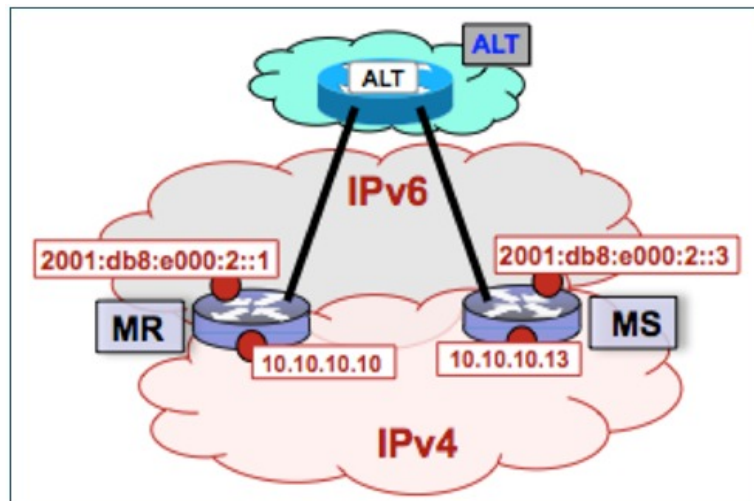
	Command or Action	Purpose
Step 9	interface <i>type number</i> Example: <pre>Router(config)# interface tunnel 191</pre>	Specifies the interface type of tunnel and the interface number and enters interface configuration mode.
Step 10	vrf forwarding <i>vrf-name</i> Example: <pre>Router(config-if)# vrf forwarding lisp</pre>	Associates a VRF instance configured in Step 2 with the tunnel interface configured in Step 9. <ul style="list-style-type: none"> • When the interface is bound to a VRF, previously configured IP addresses are removed, and the interface is disabled.
Step 11	ip address <i>ip-address mask</i> Example: <pre>Router(config-if)# ip address 192.168.1.6 255.255.255.252</pre>	Configures an IPv4 address for the tunnel interface.
Step 12	ipv6 address <i>ipv6-address/mask</i> Example: <pre>Router(config-if)# ipv6 address 2001:DB8:ffff::6/64</pre>	Configures an IPv6 address for the tunnel interface.
Step 13	tunnel source <i>interface-type interface-number</i> Example: <pre>Router(config-if)# tunnel source GigabitEthernet 0/0/0</pre>	Configures the tunnel source.
Step 14	tunnel destination <i>ipv4-address</i> Example: <pre>Router(config-if)# tunnel destination 10.10.10.13</pre>	Configures the tunnel destination IPv4 address for the tunnel interface.
Step 15	exit Example: <pre>Router(config-if)# exit</pre>	Exits interface configuration mode and enters global configuration mode.
Step 16	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 17	ipv4 map-server Example:	Enables LISP map server functionality for EIDs in the IPv4 address family.

	Command or Action	Purpose
	Router(config-router-lisp)# ipv4 map-server	
Step 18	ipv4 alt-vrf <i>vrf-name</i> Example: Router(config-router-lisp)# ipv4 alt-vrf lisp	Associates a VRF table with the LISP ALT for IPv4 EIDs. <ul style="list-style-type: none"> In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 19	ipv6 map-server Example: Router(config-router-lisp)# ipv6 map-server	Enables LISP map server functionality for EIDs in the IPv6 address family.
Step 20	ipv6 alt-vrf <i>vrf-name</i> Example: Router(config-router-lisp)# ipv6 alt-vrf lisp	Associates a VRF table with the LISP ALT for IPv6 EIDs. <ul style="list-style-type: none"> In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 21	site <i>site-name</i> Example: Router(config-router-lisp)# site Site-1	Specifies a LISP site and enters LISP site configuration mode. <p>Note A LISP site name is locally significant to the map server on which it is configured. It has no relevance anywhere else. This name is used solely as an administrative means of associating one or more EID prefixes with an authentication key and other site-related mechanisms.</p>
Step 22	eid-prefix <i>EID-prefix</i> Example: Router(config-router-lisp-site)# eid-prefix 172.16.1.0/24	Configures an IPv4 or IPv6 EID prefix associated with this LISP site. <ul style="list-style-type: none"> Repeat this step as necessary to configure additional EID prefixes under this LISP sites. In this step example, only an IPv4 EID prefix is configured but to complete the configuration, an IPv6 EID prefix must also be configured. <p>Note The LISP ETR must be configured with matching EID prefixes and an identical authentication key.</p> <p>Note Additional eid-prefix command configuration options are available. (See the <i>LISP Command Reference</i> for more details.)</p>
Step 23	authentication-key <i>key-type authentication-key</i> Example:	Configures the authentication key associated with this site. <p>Note The LISP ETR must be configured with matching EID prefixes and an identical authentication key.</p>

	Command or Action	Purpose
	<pre>Router(config-router-lisp-site)# authentication-key 0 some-key</pre>	Note The authentication-key can be configured with Type 6 encryption. (See the <i>LISP Command Reference</i> for more details.)
Step 24	exit Example: <pre>Router(config-router-lisp-site)# exit</pre>	Exits LISP site configuration mode and returns to LISP configuration mode.
Step 25	Repeat Steps 21 through 24 to configure additional LISP sites.	—
Step 26	exit Example: <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 27	router bgp <i>autonomous-system-number</i> Example: <pre>Router(config)# router bgp 65011</pre>	Enters router configuration mode for the specified routing process.
Step 28	address-family ipv4 [unicast multicast vrf <i>vrf-name</i>] Example: <pre>Router(config-router)# address-family ipv4 vrf lisp</pre>	Specifies the IPv4 address family and enters IPv4 address family configuration mode. <ul style="list-style-type: none"> • The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent commands. • In this example, the VRF table named <i>lisp</i> (created in Step 2) is associated with the BGP IPv4 VRF that carries EID prefixes in the LISP ALT.
Step 29	redistribute lisp Example: <pre>Router(config-router-af)# redistribute lisp</pre>	Redistributes EID prefixes known to LISP into BGP.
Step 30	neighbor <i>ip-address</i> remote-as <i>autonomous-system-number</i> Example: <pre>Router(config-router-af)# neighbor 192.168.1.1 remote-as 65010</pre>	Adds the IP address of the neighbor in the specified autonomous system to the IPv4 multiprotocol BGP neighbor table of the local router.
Step 31	neighbor <i>ip-address</i> activate Example:	Enables the neighbor to exchange prefixes for the IPv4 unicast address family.

	Command or Action	Purpose
	<pre>Router(config-router-af)# neighbor 192.168.1.1 activate</pre>	
Step 32	exit Example: <pre>Router(config-router-af)# exit</pre>	Exits address family configuration mode and returns to router configuration mode.
Step 33	address-family ipv6 vrf vrf-name Example: <pre>Router(config-router)# address-family ipv6 vrf lisp</pre>	Specifies the IPv6 address family and enters IPv6 address family configuration mode. <ul style="list-style-type: none"> • The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent commands. • In this example, the VRF table named <i>lisp</i> (created in Step 2) is associated with the BGP IPv6 VRF that carries EID prefixes in the LISP ALT.
Step 34	redistribute lisp Example: <pre>Router(config-router-af)# redistribute lisp</pre>	Redistributes EID prefixes known to LISP into BGP.
Step 35	neighbor ip-address remote-as autonomous-system-number Example: <pre>Router(config-router-af)# neighbor 2001:db8:ffff::1 remote-as 65010</pre>	Adds the IPv6 address of the neighbor in the specified autonomous system to the IPv6 multiprotocol BGP neighbor table of the local router.
Step 36	neighbor ip-address activate Example: <pre>Router(config-router-af)# neighbor 2001:db8:ffff::1 activate</pre>	Enables the neighbor to exchange prefixes for the IPv6 unicast address family.
Step 37	exit Example: <pre>Router(config-router-af)# exit</pre>	Exits address family configuration mode and returns to router configuration mode.
Step 38	exit Example: <pre>Router(config-router)# exit</pre>	Exits router configuration mode and returns to global configuration mode.
Step 39	ip route ipv4-prefix next-hop Example:	Configures an IPv4 static route.

	Command or Action	Purpose
	Router(config)# ip route 0.0.0.0 0.0.0.0 10.10.10.1	<ul style="list-style-type: none"> In this example, a default route to the upstream next hop for all IPv4 destinations is created.
Step 40	ipv6 route ipv6-prefix next-hop Example: Router(config)# ipv6 route ::/0 2001:db8:e000:2::f0f	Configures an IPv6 static route. <ul style="list-style-type: none"> In this example, a default route to the upstream next hop for all IPv6 destinations is created.
Step 41	exit Example: Router(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Example:*Figure 15: ALT-Connected LISP Map Resolver and Map Server, each having both an IPv4 and an IPv6 RLOC*

The example below shows the full configuration for a LISP map server including some basic IP and IPv6 configuration not included in the task table for this task:

```

!
hostname MS
!
vrf definition lisp
 rd 1:1
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
 !
no ip domain lookup

```

```

ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 no ip address
!
interface Tunnel192
 vrf forwarding lisp
 ip address 192.168.1.2 255.255.255.252
 ipv6 address 2001:db8:ffff::2/64
 tunnel source GigabitEthernet 0/0/0
 tunnel destination 10.10.10.10
!
interface GigabitEthernet 0/0/0
 description Link to SP1 (RLOC)
 ip address 10.10.10.13 255.255.255.0
 ipv6 address 2001:db8:e000:2::3/64
!
router lisp
 site Site-1
  authentication-key 0 some-xtr-key
  eid-prefix 172.16.1.0/24
  eid-prefix 2001:db8:a::/48
  exit
!
 site Site-2
  authentication-key 0 another-xtr-key
  eid-prefix 172.16.2.0/24
  eid-prefix 2001:db8:b::/48
  exit
!
!---configure more LISP sites as required---
!
ipv4 map-server
ipv4 alt-vrf lisp
ipv6 map-server
ipv6 alt-vrf lisp
exit
!
router bgp 65011
 bgp asnotation dot
 bgp log-neighbor-changes
!
 address-family ipv4 vrf lisp
  redistribute lisp
  neighbor 192.168.1.1 remote-as 65010
  neighbor 192.168.1.1 activate
 exit-address-family
!
 address-family ipv6 vrf lisp
  redistribute lisp
  neighbor 2001:db8:ffff::1 remote-as 65010
  neighbor 2001:db8:ffff::1 activate
 exit-address-family
!
ip route 0.0.0.0 0.0.0.0 10.10.10.1
!
ipv6 route ::/0 2001:db8:e000:2::f0f

```

Configure a PETR and a PITR

The following tasks show how to design and deploy a Proxy Egress Tunnel Router (PETR) and a Proxy Ingress Tunnel Router (PITR). The example scenario shows deployment of a PETR and PITR as separate devices but it is also possible to deploy a single device that acts simultaneously as a PETR and a PITR, which is called a PxTR.

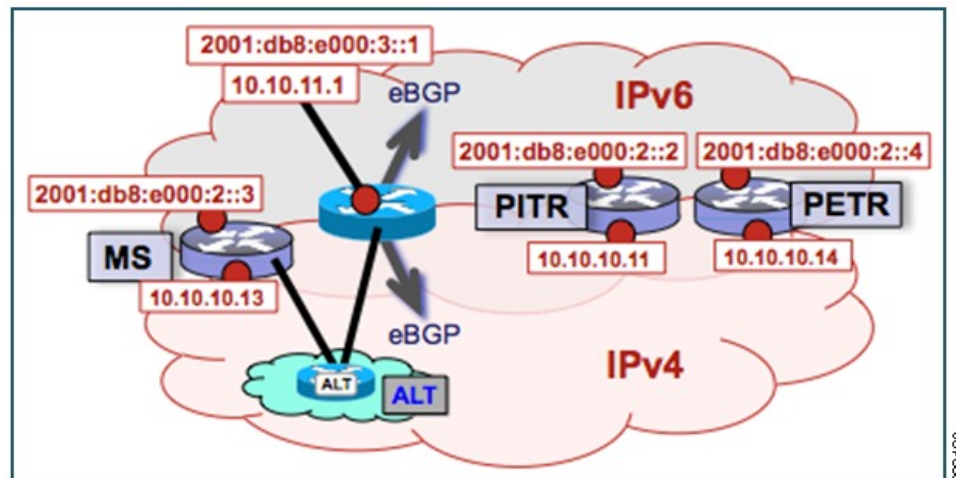
Deploying a Proxy Egress Tunnel Router with both an IPv4 and an IPv6 RLOC

Perform this task to deploy a Proxy Egress Tunnel Router (PETR) for both IPv4 and IPv6 address families. You can also perform this task to configure PETR functionality on a single device that acts simultaneously as a PETR and as a Proxy Ingress Tunnel Router (PITR), referred to as a PxTR.

A PETR simply takes in LISP encapsulated packets and decapsulates them and forwards them. For example, a PETR can be used to provide IPv6 LISP EIDs access to non-LISP EIDs when the LISP site only has IPv4 RLOC connectivity. A PETR, therefore, is used for LISP-to-non-LISP access in situations where cross-address family connectivity is an issue. (A PETR can still be used for matching EID and RLOC address families if desired.) Note that a PITR is required to provide return-traffic flow. A PETR is simple to deploy because it need only provide dual-stack connectivity to the core.

The topology used in this PETR example is shown in the figure. The PETR and PITR in this example are deployed as separate devices and each have both an IPv4 and an IPv6 locator.

Figure 16: Proxy Egress Tunnel Router with both an IPv4 and an IPv6 RLOC



The components illustrated in the topology shown in the figure are described below:

PETR

- When deployed as a standalone LISP device, the PETR has dual-stack connectivity to the core network.
- The PETR IPv4 locator is 10.10.10.14/24 and the IPv6 locator is 2001:db8:e000:2::4/64.

SUMMARY STEPS

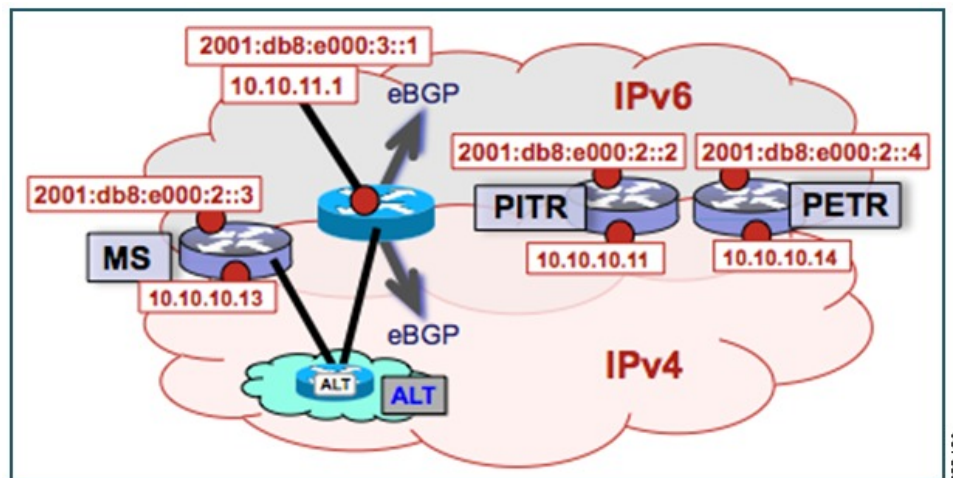
1. `enable`
2. `configure terminal`
3. `router lisp`

4. `ipv4 proxy-etr`
5. `ipv6 proxy-etr`
6. `exit`
7. `ip route ipv4-prefix next-hop`
8. `ipv6 route ipv6-prefix next-hop`
9. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 4	ipv4 proxy-etr Example: <pre>Router(config-router-lisp)# ipv4 proxy-etr</pre>	Enables PETR functionality for IPv4 EIDs.
Step 5	ipv6 proxy-etr Example: <pre>Router(config-router-lisp)# ipv6 proxy-etr</pre>	Enables PETR functionality for IPv6 EIDs.
Step 6	exit Example: <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and enters global configuration mode.
Step 7	ip route ipv4-prefix next-hop Example: <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.10.10.1</pre>	Configures an IPv4 static route. <ul style="list-style-type: none"> • In this example, a default route to the upstream next hop for all IPv4 destinations is created.
Step 8	ipv6 route ipv6-prefix next-hop Example:	Configures an IPv6 static route. <ul style="list-style-type: none"> • In this example, a default route to the upstream next hop for all IPv6 destinations is created.

	Command or Action	Purpose
	Router(config)# ipv6 route ::/0 2001:db8:e000:2::f0f	
Step 9	exit Example: Router(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Example:*Figure 17: Proxy Egress Tunnel Router with both an IPv4 and an IPv6 RLOC*

The example below shows the full configuration for a PETR including some basic IP and IPv6 configuration not included in the task table for this task:

```

!
hostname PETR
!
no ip domain lookup
ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 no ip address
!
interface GigabitEthernet 0/0/0
 description Link to Core (RLOC)
 ip address 10.10.10.14 255.255.255.0
 ipv6 address 2001:db8:e000:2::4/64
!
router lisp
 ipv4 proxy-etr
 ipv6 proxy-etr
 exit
!

```

```
ip route 0.0.0.0 0.0.0.0 10.10.10.1
!
ipv6 route ::/0 2001:db8:e000:2::f0f
```

Deploying a Proxy Ingress Tunnel Router with both an IPv4 and an IPv6 RLOC

Perform this task to deploy a Proxy Ingress Tunnel Router (PITR) for both IPv4 and IPv6 address families. You can also perform this task to configure PITR functionality on a single device that acts simultaneously as a PITR and as a Proxy Egress Tunnel Router (PETR), referred to as a PxTR.

A PITR attracts non-LISP packets by advertising a coarse-aggregate prefix for LISP EIDs into the core (such as the Internet or a Multiprotocol Label Switching (MPLS) core) and then performs LISP encapsulation services (like an ITR) to provide access to LISP EIDs. Thus, a PITR provides non-LISP-to-LISP interworking. A PITR is also used to provide address family “hop-over” for non-LISP-to-LISP traffic. For example, a dual-stacked PxTR can be used to provide a return-traffic path from non-LISP IPv6 sites to IPv6 LISP sites that contain only IPv4 RLOCs.

To resolve EID-to-RLOC mappings for creating non-LISP-to-LISP flows, configure PITR to query the LISP mapping system. In this task, the PITR is configured to send Map-Rrequest messages via the LISP alternate logical topology (ALT) to resolve EID-to-RLOC mappings.

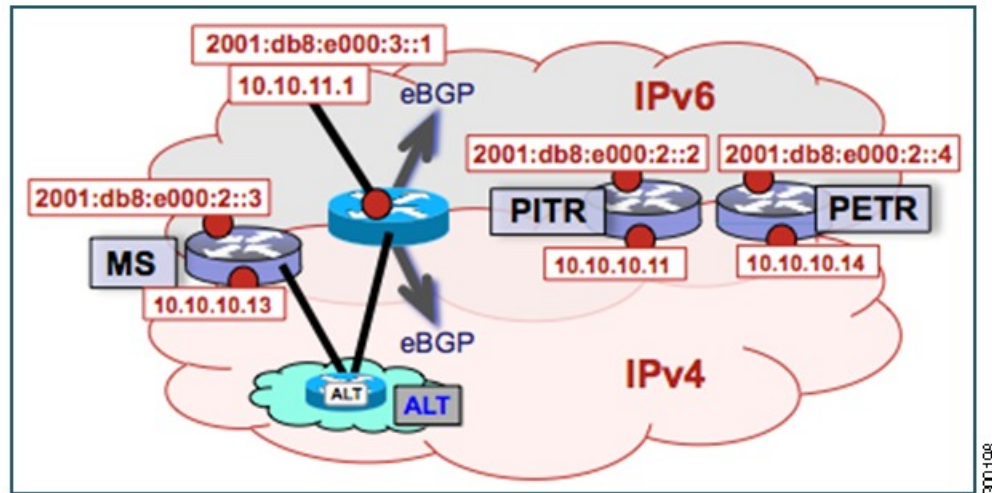


Note

To attract non-LISP traffic destined to LISP sites, the PITR must advertise coarse-aggregate EID prefixes into the underlying network infrastructure. In an Internet-as-the-core example, attracting non-LISP traffic destined to LISP sites is typically managed via external BGP (eBGP) and by advertising the coarse-aggregate that includes all appropriate EID prefixes into the Internet. The example configuration in the figure utilizes this approach. Because this is a standard BGP configuration, summary and detailed command guidance is not provided in the task table for this task, although the complete configuration example that follows the task table does include an accurate example of this eBGP peering. Any other approach that advertises coarse-aggregates that include all appropriate EID prefixes into the core are also acceptable.

The topology used in this example is shown in the figure. The PITR is deployed as a separate device, with both an IPv4 and an IPv6 locator. A map resolver and core-peering router are also shown in the figure for reference because they are required components for completing the PITR configuration shown in the figure.

Figure 18: Proxy Ingress Tunnel Router with both an IPv4 and an IPv6 RLOC



The components illustrated in the topology shown in the figure are described below:

PITR

- When deployed as a standalone LISP device, the PITR has dual-stack connectivity to the core network.
- The PITR IPv4 locator is 10.10.10.11/24 and the IPv6 locator is 2001:db8:e000:2::2/64.
- The use of LISP EID prefixes throughout this task (172.16.1.0/24 and 2001:db8:a::/48 configuration) is assumed and are part of LISP EID blocks that can be summarized in coarse-aggregates and advertised by the PITR into the core network. The advertisement of the IPv4 coarse-aggregate of 172.16.0.0/16 and the IPv6 coarse-aggregate of 2001:db8::/33 by the PITR into the IPv4 and IPv6 core networks is also assumed.
- The PITR eBGP peers with the core router with locators 10.10.11.1 and 2001:db8:e000:3::1 in order to advertise the coarse-aggregate IPv4 EID prefix of 172.16.0.0/16 and the IPv6 EID prefix of 2001:db8::/33 into the IPv4 and IPv6 cores, respectively.
- The PITR is configured to use the LISP ALT (GRE+BGP) via the map server with locators 10.10.10.13 and 2001:db8:e000:2::3. The relevant configuration is shown for the PITR.

Other Infrastructure

- The MS has IPv4 and IPv6 tunnel endpoints in the VRF table (named lisp) of 192.168.5/30 and 2001:db8:fff::5/64, respectively. The configuration of the map server is not in the task table.
- The core router has an IPv4 address of 10.10.11.1 and an IPv6 address of 2001:db8:e000:3::1. These addresses will be used for eBGP peering. The core router configuration is assumed to be familiar as a typical ISP peering router and is therefore not included in the task table.

SUMMARY STEPS

1. **configure terminal**
2. **vrf definition** *vrf-name*
3. **rd** *route-distinguisher*
4. **address-family ipv4** [unicast]

5. **exit-address-family**
6. **address-family ipv6**
7. **exit-address-family**
8. **exit**
9. **interface** *type number*
10. **vrf forwarding** *vrf-name*
11. **ip address** *ip-address mask*
12. **ipv6 address** *ipv6-address/mask*
13. **tunnel source** *interface-type interface-number*
14. **tunnel destination** *ipv4-address*
15. **exit**
16. **router lisp**
17. **ipv4 alt-vrf** *vrf-name*
18. **ipv4 proxy-itr** *ipv4-locator [ipv6-locator]*
19. **ipv4 map-cache-limit** *map-cache-limit*
20. **ipv6 alt-vrf** *vrf-name*
21. **ipv6 proxy-itr** *ipv6-locator [ipv4-locator]*
22. **ipv6 map-cache-limit** *map-cache-limit*
23. **exit**
24. **router bgp** *autonomous-system-number*
25. **address-family ipv4** [**unicast** | **multicast** | **vrf** *vrf-name*]
26. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
27. **neighbor** *ip-address* **activate**
28. **exit**
29. **address-family ipv6** [**unicast** | **multicast** | **vrf** *vrf-name*]
30. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
31. **neighbor** *ip-address* **activate**
32. **exit**
33. **exit**
34. **ip route** *ipv4-prefix next-hop*
35. **ip route** *ipv4-prefix next-hop*
36. **ipv6 route** *ipv6-prefix next-hop*
37. **ipv6 route** *ipv6-prefix next-hop*
38. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 2	vrf definition <i>vrf-name</i> Example:	Configures a virtual routing and forwarding (VRF) table and enters VRF configuration mode.

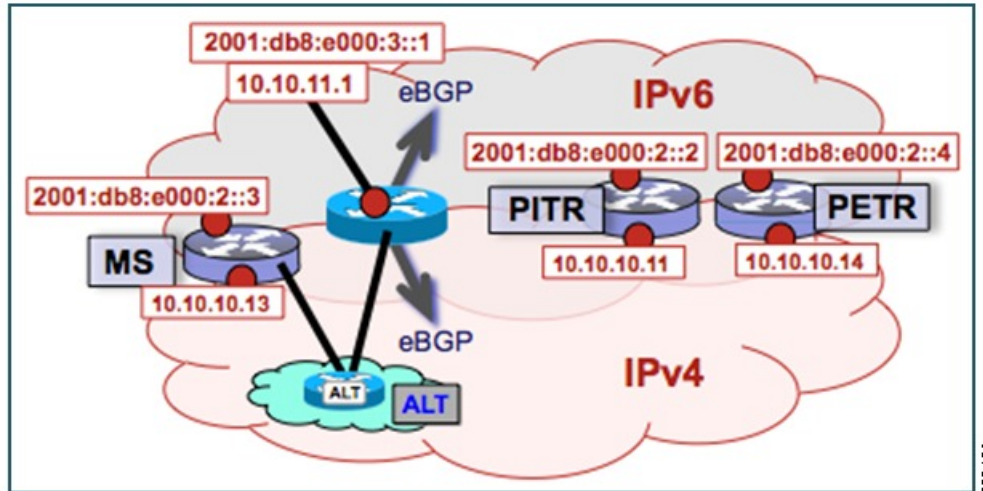
	Command or Action	Purpose
	<code>Router(config)# vrf definition lisp</code>	<ul style="list-style-type: none"> Use the <i>vrf-name</i> argument to specify a name to be assigned to the VRF. In this example, a VRF named <code>lisp</code> is created to hold EID prefixes.
Step 3	rd <i>route-distinguisher</i> Example: <code>Router(config-vrf)# rd 1:1</code>	Creates routing and forwarding tables for a VRF.
Step 4	address-family ipv4 [unicast] Example: <code>Router(config-vrf)# address-family ipv4</code>	Enters VRF IPv4 address family configuration mode to specify an IPv4 address family for a VRF table. <ul style="list-style-type: none"> In this example, the VRF named <code>lisp</code> handles IPv4 EID prefixes.
Step 5	exit-address-family Example: <code>Router(config-vrf-af)# exit-address-family</code>	Exits VRF address family configuration mode and returns to VRF configuration mode.
Step 6	address-family ipv6 Example: <code>Router(config-vrf)# address-family ipv6</code>	Enters VRF IPv6 address family configuration mode to specify an IPv6 address family for a VRF table. <ul style="list-style-type: none"> In this example, the VRF table named <code>lisp</code> handles IPv6 EID prefixes.
Step 7	exit-address-family Example: <code>Router(config-vrf-af)# exit-address-family</code>	Exits VRF address family configuration mode and returns to VRF configuration mode.
Step 8	exit Example: <code>Router(config-vrf)# exit</code>	Exits VRF configuration mode and enters global configuration mode.
Step 9	interface <i>type number</i> Example: <code>Router(config)# interface tunnel 191</code>	Specifies the interface type of tunnel and the interface number and enters interface configuration mode.
Step 10	vrf forwarding <i>vrf-name</i> Example: <code>Router(config-if)# vrf forwarding lisp</code>	Associates a VRF instance configured in Step 2 with the tunnel interface configured in Step 9. <ul style="list-style-type: none"> When the interface is bound to a VRF, previously configured IP addresses are removed, and the interface is disabled.

	Command or Action	Purpose
Step 11	ip address <i>ip-address mask</i> Example: <pre>Router(config-if)# ip address 192.168.1.6 255.255.255.252</pre>	Configures an IPv4 address for the tunnel interface.
Step 12	ipv6 address <i>ipv6-address/mask</i> Example: <pre>Router(config-if)# ipv6 address 2001:DB8:ffff::6/64</pre>	Configures an IPv6 address for the tunnel interface.
Step 13	tunnel source <i>interface-type interface-number</i> Example: <pre>Router(config-if)# tunnel source GigabitEthernet 0/0/0</pre>	Configures the tunnel source.
Step 14	tunnel destination <i>ipv4-address</i> Example: <pre>Router(config-if)# tunnel destination 10.10.10.13</pre>	Configures the tunnel destination IPv4 address for the tunnel interface.
Step 15	exit Example: <pre>Router(config-if)# exit</pre>	Exits interface configuration mode and enters global configuration mode.
Step 16	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 17	ipv4 alt-vrf <i>vrf-name</i> Example: <pre>Router(config-router-lisp)# ipv4 alt-vrf lisp</pre>	Associates a VRF table with the LISP ALT for IPv4 EIDs. <ul style="list-style-type: none"> • In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 18	ipv4 proxy-itr <i>ipv4-locator [ipv6-locator]</i> Example: <pre>Router(config-router-lisp)# ipv4 proxy-itr 10.10.10.11 2001:db8:e000:2::2</pre>	Enables Proxy Ingress Tunnel Router (PITR) functionality for IPv4 EIDs, and specifies the IPv4 and (optionally) the IPv6 RLOCs (local to the PITR) to use when LISP-encapsulating packets to LISP sites.
Step 19	ipv4 map-cache-limit <i>map-cache-limit</i> Example:	Specifies the maximum number of IPv4 map-cache entries to be maintained by the PITR. <ul style="list-style-type: none"> • When the map-cache reaches this limit, existing entries are removed according to the rules described

	Command or Action	Purpose
	<pre>Router(config-router-lisp)# ipv4 map-cache-limit 100000</pre>	<p>in the command reference guide. (See the <i>LISP Command Reference</i> for more details.)</p> <ul style="list-style-type: none"> The default map-cache-limit is 10000. In this example, since the device is being configured as a PITR, a larger map-cache limit is configured.
Step 20	<p>ipv6 alt-vrf <i>vrf-name</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 alt-vrf lisp</pre>	<p>Associates a VRF table with the LISP ALT for IPv6 EIDs.</p> <ul style="list-style-type: none"> In this example, the VRF table named lisp (created in Step 2) is associated with the LISP ALT.
Step 21	<p>ipv6 proxy-itr <i>ipv6-locator [ipv4-locator]</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 proxy-itr 2001:db8:e000:2::2 10.10.10.11</pre>	<p>Enables Proxy Ingress Tunnel Router (PITR) functionality for IPv6 EIDs, and specifies the IPv6 and (optionally) the IPv4 RLOCs (local to the PITR) to use when LISP-encapsulating packets to LISP sites.</p>
Step 22	<p>ipv6 map-cache-limit <i>map-cache-limit</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 map-cache-limit 100000</pre>	<p>Specifies the maximum number of IPv6 map-cache entries to be maintained by the PITR.</p> <ul style="list-style-type: none"> When the map-cache reaches this limit, existing entries are removed according to the rules described in the command reference guide. (See the <i>LISP Command Reference</i> for more details.) <p>The default map-cache-limit is 10000. In this example, since the device is being configured as a PITR, a larger map-cache limit is configured.</p>
Step 23	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	<p>Exits LISP configuration mode and returns to global configuration mode.</p>
Step 24	<p>router bgp <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Router(config)# router bgp 65015</pre>	<p>Enters router configuration mode for the specified routing process.</p>
Step 25	<p>address-family ipv4 [unicast multicast vrf <i>vrf-name</i>]</p> <p>Example:</p> <pre>Router(config-router)# address-family ipv4 vrf lisp</pre>	<p>Specifies the IPv4 address family and enters IPv4 address family configuration mode.</p> <ul style="list-style-type: none"> The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent commands. In this example, the VRF table named lisp (created in Step 2) is associated with the BGP IPv4 VRF that carries EID prefixes in the LISP ALT.

	Command or Action	Purpose
Step 26	<p>neighbor <i>ip-address</i> remote-as <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 192.168.1.5 remote-as 65011</pre>	Adds the IP address of the neighbor in the specified autonomous system to the IPv4 multiprotocol BGP neighbor table of the local router.
Step 27	<p>neighbor <i>ip-address</i> activate</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 192.168.1.5 activate</pre>	Enables the neighbor to exchange prefixes for the IPv4 unicast address family.
Step 28	<p>exit</p> <p>Example:</p> <pre>Router(config-router-af)# exit</pre>	Exits address family configuration mode.
Step 29	<p>address-family ipv6 [unicast multicast vrf <i>vrf-name</i>]</p> <p>Example:</p> <pre>Router(config-router-af)# address-family ipv6 vrf lisp</pre>	<p>Specifies the IPv6 address family and enters IPv6 address family configuration mode.</p> <ul style="list-style-type: none"> The vrf keyword and <i>vrf-name</i> argument specify the name of the VRF instance to associate with subsequent commands. In this example, the VRF table named <i>lisp</i> (created in Step 2) is associated with the BGP IPv6 VRF that carries EID prefixes in the LISP ALT.
Step 30	<p>neighbor <i>ip-address</i> remote-as <i>autonomous-system-number</i></p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 2001:db8:ffff::5 remote-as 65011</pre>	Adds the IPv6 address of the neighbor in the specified autonomous system to the IPv6 multiprotocol BGP neighbor table of the local router.
Step 31	<p>neighbor <i>ip-address</i> activate</p> <p>Example:</p> <pre>Router(config-router-af)# neighbor 2001:db8:ffff::5 activate</pre>	Enables the neighbor to exchange prefixes for the IPv6 unicast address family.
Step 32	<p>exit</p> <p>Example:</p> <pre>Router(config-router-af)# exit</pre>	Exits address family configuration mode.
Step 33	<p>exit</p> <p>Example:</p>	Exits router configuration mode.

	Command or Action	Purpose
	<code>Router(config-router)# exit</code>	
Step 34	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.10.10.1</pre>	<p>Configures an IPv4 static route.</p> <ul style="list-style-type: none"> In this example, a default route to the upstream next hop for all IPv4 destinations is created.
Step 35	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 172.16.0.0 255.255.0.0 Null0 tag 123</pre>	<p>Configures an IPv4 static route.</p> <ul style="list-style-type: none"> In this example, a static route is configured to Null0 for the coarse-aggregate IPv4 EID prefix 172.16.0.0/16. This static route is required to ensure proper operation of LISP in querying the mapping system for LISP EIDs. The tag 123 is added to this null route as a reference point for the route map used to permit the advertisement of this coarse aggregate to the upstream ISP BGP peer.
Step 36	<p>ipv6 route <i>ipv6-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ipv6 route ::/0 2001:db8:e000:2::f0f</pre>	<p>Configures an IPv6 static route.</p> <ul style="list-style-type: none"> In this example, a default route to the upstream next hop for all IPv6 destinations is created.
Step 37	<p>ipv6 route <i>ipv6-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ipv6 route 2001:db8::/33 Null0 tag 123</pre>	<p>Configures an IPv6 static route.</p> <ul style="list-style-type: none"> In this example, a static route is configured to Null0 for the coarse-aggregate IPv6 EID prefix 2001:db8::/33. This is required to ensure proper operation of LISP in querying the mapping system for LISP EIDs. The tag 123 is added to this null route as a handy reference point for the route-map used to permit the advertisement of this coarse-aggregate to the upstream ISP BGP peer.
Step 38	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	Exits global configuration mode.

Example:**Figure 19: Proxy Ingress Tunnel Router with both an IPv4 and an IPv6 RLOC**

The example below shows the full configuration for a PITR includes some basic IP, BGP, and route map configuration not included in the task table for this task:

```

!
hostname PITR
!
no ip domain lookup
ip cef
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 no ip address
!
interface Tunnel191
 vrf forwarding lisp
 ip address 192.168.1.6 255.255.255.252
 ipv6 address 2001:db8:ffff::6/64
 tunnel source GigabitEthernet 0/0/0
 tunnel destination 10.10.10.13
!
interface GigabitEthernet 0/0/0
 description Link to Core (RLOC)
 ip address 10.10.10.11 255.255.255.0
 ipv6 address 2001:db8:e000:2::2/64
!
router lisp
 ipv4 alt-vrf lisp
 ipv4 map-cache-limit 100000
 ipv4 proxy-itr 10.10.10.11 2001:db8:e000:2::2
 ipv6 alt-vrf lisp
 ipv6 map-cache-limit 100000
 ipv6 proxy-itr 2001:db8:e000:2::2 10.10.10.11
 exit
!
router bgp 65015
 bgp asnotation dot
 bgp log-neighbor-changes

```

```

neighbor 10.10.11.1 remote-as 65111
neighbor 2001:db8:e000:3::1 remote-as 65111
!
address-family ipv4
  no synchronization
  redistribute static route-map populate-default
  neighbor 10.10.11.1 activate
  neighbor 10.10.11.1 send-community both
  neighbor 10.10.11.1 route-map dfz-out out
exit-address-family
!
address-family ipv6
  redistribute static route-map populate-default
  neighbor 2001:db8:e000:3::1 activate
  neighbor 2001:db8:e000:3::1 send-community both
  neighbor 2001:db8:e000:3::1 route-map dfz-out out
exit-address-family
!
address-family ipv4 vrf lisp
  no synchronization
  neighbor 192.168.1.5 remote-as 65011
  neighbor 192.168.1.5 activate
exit-address-family
!
address-family ipv6 vrf lisp
  no synchronization
  neighbor 2001:db8:ffff::5 remote-as 65011
  neighbor 2001:db8:ffff::5 activate
exit-address-family
!
ip bgp-community new-format
ip community-list standard dfz-upstream permit 65100:123
!
ip route 0.0.0.0 0.0.0.0 10.10.10.1
ip route 172.16.0.0 255.255.0.0 Null0 tag 123
!
ipv6 route 2001:db8::/33 Null0 tag 123
ipv6 route ::/0 2001:db8:e000:2::f0f
!
route-map populate-default permit 10
  match tag 123
  set origin igp
  set community 65100:123
!
route-map dfz-out permit 10
  match community dfz-upstream
!

```

Verify and Troubleshoot Locator ID Separation Protocol

Once LISP is configured, you can verify and troubleshoot LISP configuration and operations by following the optional steps in this task. Note that certain verification and troubleshooting steps are specific to certain LISP devices and only apply if configured in your LISP site. For the below commands, if **instance-id** is unspecified, the ID is specified as zero.

SUMMARY STEPS

1. **enable**
2. **show running-config | section router lisp**

3. **show** [ip | ipv6] lisp [instance-id *number*]
4. **show** [ip | ipv6] lisp map-cache [instance-id *number*]
5. **show** [ip | ipv6] lisp database [instance-id *number*]
6. **show** lisp site [name *site-name*]
7. **lig** {[instance-id *number*] [self {ipv4 | ipv6}]} | {hostname | destination-EID}}
8. **ping** {hostname | destination-EID}
9. **clear** [ip | ipv6] lisp map-cache [instance-id *number* | *]

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 show running-config | section router lisp

The **show running-config | section router lisp** command is useful for quickly verifying the LISP configuration on the device. This command applies to any Cisco IOS LISP device.

The following is sample output from the **show running-config | section router lisp** command when a multithomed LISP site is configured with IPv4 and IPv6 EID prefixes:

Example:

```
Router# show running-config | section router lisp
router lisp
  service ipv4
    itr map-resolver 10.10.10.10
    itr map-resolver 10.10.30.10
    itr
    etr map-server 10.10.10.10 key some-key
    etr map-server 10.10.30.10 key some-key
    etr
    exit-service-ipv4
  !
  service ipv6
    itr map-resolver 10.10.10.10
    itr map-resolver 10.10.30.10
    itr
    etr map-server 10.10.10.10 key some-key
    etr map-server 10.10.30.10 key some-key
    etr
    use-petr 10.10.10.11
    use-petr 10.10.30.11
    exit-service-ipv6
  !
  instance-id 1
    service ipv4
      eid-table default
      database-mapping 172.16.1.0/24 10.1.1.2 priority 1 weight 50
      database-mapping 172.16.1.0/24 10.2.1.2 priority 1 weight 50
      exit-service-ipv4
    !
    service ipv6
      eid-table default
      database-mapping 2001:DB8:A::/48 10.1.1.2 priority 1 weight 50
```



```

    database-mapping 2001:DB8:A::/48 10.2.1.2 priority 1 weight 50
  exit-service-ipv6
  !
  exit-instance-id
  !
  exit-router-lisp

```

Step 3 **show [ip | ipv6] lisp [instance-id *number*]**

The **show ip lisp** and **show ipv6 lisp** commands are useful for quickly verifying the operational status of LISP as configured on the device, as applicable to the IPv4 and IPv6 address families, respectively. This command applies to any Cisco IOS LISP device.

Example:

The following example shows LISP operational status and IPv4 address family information:

```
Router# show ip lisp
```

```

Ingress Tunnel Router (ITR):      enabled
Egress Tunnel Router (ETR):      enabled
Proxy-ITR Router (PITR):        disabled
Proxy-ETR Router (PETR):        disabled
Map Server (MS):                 disabled
Map Resolver (MR):              disabled
Map-Request source:             172.16.1.1
ITR Map-Resolver(s):            10.10.10.10, 10.10.30.10
ETR Map-Server(s):              10.10.10.10 (00:00:56), 10.10.30.10 (00:00:12)
ETR accept mapping data:        disabled, verify disabled
ETR map-cache TTL:              1d00h
Locator Status Algorithms:
  RLOC-probe algorithm:          disabled
Static mappings configured:      0
Map-cache size/limit:           2/1000
Map-cache activity check period: 60 secs
Map-database size:              1

```

Example:

The following example shows LISP operational status and IPv6 address family information:

```
Router# show ip lisp
```

```

Ingress Tunnel Router (ITR):      enabled
Egress Tunnel Router (ETR):      enabled
Proxy-ITR Router (PITR):        disabled
Proxy-ETR Router (PETR):        disabled
Map Server (MS):                 disabled
Map Resolver (MR):              disabled
Map-Request source:             2001:DB8:A::1
ITR Map-Resolver(s):            10.10.10.10, 10.10.30.10
ETR Map-Server(s):              10.10.10.10 (00:00:23), 10.10.30.10 (00:00:40)
ETR accept mapping data:        disabled, verify disabled
ETR map-cache TTL:              1d00h
Locator Status Algorithms:
  RLOC-probe algorithm:          disabled
Static mappings configured:      0
Map-cache size/limit:           1/1000
Map-cache activity check period: 60 secs
Map-database size:              1

```

Step 4 **show [ip | ipv6] lisp map-cache [instance-id *number*]**

The **show ip lisp map-cache** and **show ipv6 lisp map-cache** commands are useful for quickly verifying the operational status of the map-cache on a device configured as an ITR or Pitr, as applicable to the IPv4 and IPv6 address families, respectively. Based on a configuration when a multihomed LISP site is configured with IPv4 and IPv6 EID prefixes, this example output assumes that a map-cache entry has been received for another site with the IPv4 EID prefix of 172.16.2.0/24 and the IPv6 EID prefix of 2001:db8:b::/48.

Example:

The following example shows IPv4 mapping cache information:

```
Router# show ip lisp map-cache

LISP IPv4 Mapping Cache, 2 entries

0.0.0.0/0, uptime: 02:48:19, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
172.16.2.0/24, uptime: 01:45:24, expires: 22:14:28, via map-reply, complete
  Locator   Uptime   State    Pri/Wgt
  10.0.0.6  01:45:24 up        1/1
```

Example:

The following example shows IPv6 mapping cache information:

```
Router# show ipv6 lisp map-cache

LISP IPv6 Mapping Cache, 2 entries

::/0, uptime: 02:49:39, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
2001:DB8:B::/48, uptime: 00:00:07, expires: 23:59:46, via map-reply, complete
  Locator   Uptime   State    Pri/Wgt
  10.0.0.6  00:00:07 up        1/1
```

Step 5 **show [ip | ipv6] lisp database [instance-id number]**

The **show ip lisp database** and **show ipv6 lisp database** commands are useful for quickly verifying the the operational status of the database mapping on a device configured as an ETR, as applicable to the IPv4 and IPv6 address families, respectively. The following example output is based on a configuration when a multihomed LISP site is configured with IPv4 and IPv6 EID prefixes.

Example:

The following example shows IPv4 mapping database information:

```
Router# show ip lisp database

LISP ETR IPv4 Mapping Database, LSBs: 0x3, 1 entries

172.16.1.0/24
```

Example:

The following example shows IPv6 mapping database information:

```
Router# show ipv6 lisp database

LISP ETR IPv6 Mapping Database, LSBs: 0x1, 1 entries

2001:DB8:A::/48
```

Step 6 **show lisp site [name site-name]**

The **show lisp site** command is useful for quickly verifying the operational status of LISP sites, as configured on a map server. This command applies only to a device configured as a map server.

The following examples are based on configurations where a multihomed LISP site is configured with both IPv4 and IPv6 EID prefixes:

Example:

```
Router# show lisp site
```

```
LISP Site Registration Information
```

Site Name	Last Register	Up	Who Registered	EID Prefix
Site-1	00:00:15	yes	10.1.1.2	172.16.1.0/24
	00:00:11	yes	10.1.1.2	2001:DB8:A::/48
Site-2	00:00:27	yes	10.0.0.6	172.16.2.0/24
	00:00:37	yes	10.0.0.6	2001:DB8:B::/48

Example:

```
Router# show lisp site name Site-1
```

```
Site name: Site-1
Allowed configured locators: any
Allowed EID-prefixes:
  EID-prefix: 172.16.1.0/24
    First registered:    00:04:51
    Routing table tag:  0
    Origin:             Configuration
    Merge active:       No
    Proxy reply:        No
    TTL:                1d00h
  Registration errors:
    Authentication failures: 0
    Allowed locators mismatch: 0
  ETR 10.1.1.2, last registered 00:00:01, no proxy-reply, map-notify
    TTL 1d00h, no merge
    Locator Local State Pri/Wgt
    10.1.1.2 yes up 1/50
  ETR 10.2.1.2, last registered 00:00:03, no proxy-reply, map-notify
    TTL 1d00h, merge
    Locator Local State Pri/Wgt
    10.1.1.2 yes up 1/50
    10.2.1.2 yes up 1/50
  EID-prefix: 2001:DB8:A::/48
    First registered:    00:04:51
    Routing table tag:  0
    Origin:             Configuration
    Merge active:       No
    Proxy reply:        No
    TTL:                1d00h
  Registration errors:
    Authentication failures: 0
    Allowed locators mismatch: 0
  ETR 10.1.1.2, last registered 00:00:01, no proxy-reply, map-notify
    TTL 1d00h, no merge
    Locator Local State Pri/Wgt
    10.1.1.2 yes up 1/50
  ETR 10.2.1.2, last registered 00:00:03, no proxy-reply, map-notify
    TTL 1d00h, merge
    Locator Local State Pri/Wgt
    10.1.1.2 yes up 1/50
    10.2.1.2 yes up 1/50
```

Step 7 **lig** `{[instance-id number] [self {ipv4 | ipv6}] | {hostname | destination-EID}}`

The LISP Internet Groper (**lig**) command is useful for testing the LISP control plane. The **lig** command can be used to query for the indicated destination hostname or EID, or the router's local EID prefix. This command provides a simple means of testing whether a destination EID exists in the LISP mapping database system, or whether your site is registered with the mapping database system. This command is applicable for both the IPv4 and IPv6 address families and applies to any Cisco IOS LISP device that maintains a map-cache (i.e. configured as an ITR or Pitr).

The following examples are based on configurations where a multihomed LISP site is configured with both IPv4 and IPv6 EID prefixes:

Example:

```
Router# lig self ipv4
```

```
Mapping information for EID 172.16.1.0 from 10.1.1.2 with RTT 12 msec
172.16.1.0/24, uptime: 00:00:00, expires: 23:59:52, via map-reply, self
  Locator  Uptime  State  Pri/Wgt
  10.1.1.2 00:00:00 up, self 1/50
  10.2.1.2 00:00:00 up      1/50
```

Example:

```
Router# lig self ipv6
```

```
Mapping information for EID 2001:DB8:A:: from 10.0.0.2 with RTT 12 msec
2001:DB8:A::/48, uptime: 00:00:00, expires: 23:59:52, via map-reply, self
  Locator  Uptime  State  Pri/Wgt
  10.1.1.2 00:00:00 up, self 1/50
  10.2.1.2 00:00:00 up      1/50
```

Example:

```
Router# lig 172.16.2.1
```

```
Mapping information for EID 2001:DB8:A:: from 10.0.0.2 with RTT 12 msec
2001:DB8:A::/48, uptime: 00:00:00, expires: 23:59:52, via map-reply, self
  Locator  Uptime  State  Pri/Wgt
  10.1.1.2 00:00:00 up, self 1/50
  10.2.1.2 00:00:00 up      1/50
```

Example:

```
Router# lig 2001:db8:b::1
```

```
Mapping information for EID 172.16.2.1 from 10.0.0.6 with RTT 4 msec
2001:DB8:B::/48, uptime: 01:52:45, expires: 23:59:52, via map-reply, complete
  Locator  Uptime  State  Pri/Wgt
  10.0.0.6 01:52:45 up      1/1
```

Step 8 **ping** `{hostname | destination-EID}`

The **ping** command is useful for testing basic network connectivity and reachability and liveness of a destination EID or RLOC address. It is important to be aware that because LISP uses encapsulation, you should always specify a source address when using **ping**. Never allow the **ping** application to assign its own default source address because there are four possible ways to use **ping** and unless the source address is explicitly named, the wrong address may be used by the application and return erroneous results that complicate operational verification or troubleshooting.

The four possible uses of **ping** are:

- RLOC-to-RLOC—Sends out “echo” packets natively (no LISP encapsulation) and receives the “echo-reply” back natively. This use of **ping** can test the underlying network connectivity between locators of various devices, such as between an xTR and a map server or map resolver.

- EID-to-EID—Sends out “echo” packets with LISP encapsulation and receives the “echo-reply” back as LISP encapsulated. This use of **ping** can be used to test the LISP data plane (encapsulation) between LISP sites.
- EID-to-RLOC—Sends out “echo” packets natively (no LISP encapsulation) and receives the “echo-reply” back as LISP encapsulated through a Pitr mechanism. This use of **ping** can be used to test the Pitr infrastructure.
- RLOC-to-EID - Sends out “echo” packets with LISP encapsulation and receives the “echo-reply” back natively (no LISP encapsulation. This use of **ping** can be used to test Pitr capabilities.

The **ping** command is applicable to the IPv4 and IPv6 address families, respectively, and can be used on any LISP device but is limited by the LISP device and site configuration. (For example, the ability to do LISP encapsulation requires the device to be configured as either an ITR or Pitr.)

The following examples are based on configurations where a multihomed LISP site is configured with both IPv4 and IPv6 EID prefixes:

Example:

```
Router# ping 172.16.2.1 source 172.16.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.2.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.1.1
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/8 ms
```

Example:

```
Router# ping 2001:db8:b::1 source 2001:db8:a::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:B::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:A::1
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/8 ms
```

Step 9 **clear [ip | ipv6] lisp map-cache [instance-id number | *]**

The **clear ip lisp map-cache** and **clear ipv6 lisp map-cache** commands remove all IPv4 or IPv6 dynamic LISP map-cache entries stored by the router. The * keyword clears all entries. This command applies to a LISP device that maintains a map-cache (like one configured as an ITR or Pitr) and can be useful if trying to quickly verify the operational status of the LISP control plane. Based on a configuration when a multihomed LISP site is configured with both IPv4 and IPv6 EID prefixes, the following example output assumes that a map-cache entry has been received for another site with the IPv4 EID prefix of 172.16.2.0/24 or an IPv6 EID prefix of 2001:db8:b::/48.

Example:

The following example shows IPv4 mapping cache information, how to clear the mapping cache, and the **show** information after the cache is cleared.

```
Router# show ip lisp map-cache

LISP IPv4 Mapping Cache, 2 entries

0.0.0.0/0, uptime: 02:48:19, expires: never, via static send map-request
Negative cache entry, action: send-map-request
172.16.2.0/24, uptime: 01:45:24, expires: 22:14:28, via map-reply, complete
Locator  Uptime   State   Pri/Wgt
10.0.0.6  01:45:24  up      1/1

Router# clear ip lisp map-cache

Router# show ip lisp map-cache
```

```
LISP IPv4 Mapping Cache, 1 entries
0.0.0.0/0, uptime: 00:00:02, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
```

Example:

The following example shows IPv6 mapping cache information, how to clear the mapping cache, and the **show** information after the cache is cleared.

```
Router# show ipv6 lisp map-cache

LISP IPv6 Mapping Cache, 2 entries

::/0, uptime: 02:49:39, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
2001:DB8:B::/48, uptime: 00:00:07, expires: 23:59:46, via map-reply, complete
  Locator   Uptime   State     Pri/Wgt
  10.0.0.6  00:00:07  up        1/1

Router# clear ip lisp map-cache

Router# show ip lisp map-cache

LISP IPv6 Mapping Cache, 1 entries

::/0, uptime: 00:00:02, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
```

Additional References for Configuring LISP

The following sections provide references related to the Locator ID Separation Protocol.

Related Documents

Document Title	Location
Cisco IOS LISP Lab Test Configuration Application Note	http://lisp4.cisco.com/lisp_tech.html
Cisco IOS IP Routing: LISP Command Reference	http://www.cisco.com/en/US/docs/ios-xml/ios/iproute_lisp/command/ip-lisp-cr-book.html

MIBs

MIB	MIBs Link
LISP MIB	To locate and download MIBs for selected platforms, Cisco IOS software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 6830	Locator/ID Separation Protocol (LISP) http://tools.ietf.org/html/draft-ietf-lisp-07
RFC 6832	Interworking LISP with IPv4 and IPv6 https://tools.ietf.org/html/rfc6832
RFC 6833	LISP Map Server https://tools.ietf.org/html/rfc6833
RFC 6835	LISP Internet Groper (LIG) https://tools.ietf.org/html/rfc6835

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 2: Feature Information for Locator/ID Separation Protocol

Feature Name	Release	Feature Configuration Information
Configure LISP		Introduces LISP functionality to support ITR, ETR, PITR, PETR, MS, MR, and LISP ALT devices for IPv4 and IPv6 address families.

Feature Name	Release	Feature Configuration Information
LISP MIB		This feature introduces LISP MIB on Cisco software.



CHAPTER 4

LISP Multicast

The LISP Multicast feature introduces support for carrying multicast traffic over a Locator ID Separation Protocol (LISP) overlay. This support currently allows for unicast transport of multicast traffic with head-end replication at the root ingress tunnel router (ITR) site. This allows network operators to use LISP to carry multicast traffic over core networks that do not have native multicast capabilities.

- [Finding Feature Information, on page 87](#)
- [Prerequisites for LISP Multicast, on page 87](#)
- [Restrictions for LISP Multicast, on page 88](#)
- [Information About LISP Multicast, on page 88](#)
- [How to Configure LISP Multicast, on page 89](#)
- [Verifying LISP Multicast, on page 93](#)
- [Configuration Examples for LISP Multicast, on page 95](#)
- [Additional References for LISP Multicast, on page 101](#)
- [Feature Information for LISP Multicast, on page 103](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for LISP Multicast

- You must configure basic LISP services on the device. Basic LISP configurations are covered in "Configuring Basic LISP" section of this configuration guide.
- You must configure IPv6 multicast and LISP services on the device. The configuration of IPv6 multicast over LISP is covered in "How to Configure LISP Multicast" and "Example: Configuring IPv6 Multicast over LISP" sections of this guide.

Restrictions for LISP Multicast

- LISP multicast does not support IPv6 endpoint identifiers (EIDs) or IPv6 routing locators (RLOCs). Only IPv4 EIDs and IPv4 RLOCs are supported.
- LISP multicast does not support Dense Mode or Bidirectional Protocol Independent Multicast (PIM). Only PIM-Sparse Mode (SM) and PIM Source Specific Multicast (SSM) modes are supported.
- LISP multicast does not support group to Rendezvous Point (RP) mapping distribution mechanisms, Auto-RP and Bootstrap Router (BSR). Only static-RP configuration is supported.
- LISP multicast does not support LISP Virtual Machine Mobility (VM-Mobility) deployment. That is, LISP multicast cannot be used as a data center interconnect (DCI) mechanism.
- IPv6 LISP multicast does not support IPv6 routing locators. Additionally, it does not support multicast transport.



Note IPv6 LISP multicast is supported only from Cisco IOS Release 16.2 onwards, though releases earlier than 16.2 supports only IPv4 LISP multicast

Information About LISP Multicast

The implementation of LISP multicast includes the following features:

- Mapping of multicast source addresses as LISP endpoint identifiers (EIDs). (Destination group addresses are not topology dependent).
- Building the multicast distribution tree across LISP overlays.
- Unicast head-end replication of multicast data packets from sources within a root ingress tunnel router (ITR) site to receiver egress tunnel routers (ETRs).
- Support for ASM (Any Source Multicast) and SSM (Source Specific Multicast).
- Support for various combinations of LISP and non-LISP capable source and receiver sites.
- Support for IPv6 endpoint identifiers (EIDs).



Note If a LISP xTR is also a PIM First Hop Router (FH) or a Rendezvous Point (RP) and the device is only receiving traffic, ensure that at least one interface on the device is covered by a local LISP database mapping. No additional configuration is required to ensure that proper address is selected.

How to Configure LISP Multicast

Configuring LISP Multicast

Perform this task to enable the LISP multicast functionality on the xTR.

Before you begin

Ensure that generic multicast functionality has been enabled on the required devices of the LISP site and PIM sparse mode has been enabled on the required interfaces of these devices.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing [distributed]**
4. Enter one of the following:
 - **ip pim rp-address** *rp-address*
 - **ip pim ssm** {**default** | **range** {*access-list-number* | *access-list-name*}}
5. **interface lisp** *interface-number*
6. **ipv6 pim lisp transport** [ipv4]
7. **ip pim sparse-mode**
8. **exit**
9. **interface** *interface-type* *interface-number*
10. **description** *string*
11. **ip pim sparse-mode**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip multicast-routing [distributed] Example: Device(config)# ip multicast-routing	Enables IP multicast routing.

	Command or Action	Purpose
Step 4	Enter one of the following: <ul style="list-style-type: none"> • <code>ip pim rp-address rp-address</code> • <code>ip pim ssm {default range {access-list-number access-list-name}}</code> Example: <pre>Device(config)# ip pim rp-address 10.1.0.2</pre> Example: <pre>Device(config)# ip pim ssm default</pre>	<ul style="list-style-type: none"> • Statically configures the address of a Protocol Independent Multicast (PIM) rendezvous point (RP) for multicast groups. • Defines the Source Specific Multicast (SSM) range of IP multicast addresses.
Step 5	interface lisp interface-number Example: <pre>Device(config)# interface LISPO</pre>	Selects a LISP interface to configure and enters interface configuration mode.
Step 6	ipv6 pim lisp transport [ipv4] Example: <pre>Device(config-if)# ipv6 pim lisp transport unicast ipv4</pre>	Selects a LISP interface to configure and enters interface configuration mode.
Step 7	ip pim sparse-mode Example: <pre>Device(config-if)# ip pim sparse-mode</pre>	Enables Protocol Independent Multicast (PIM) on an interface for sparse-mode operation.
Step 8	exit Example: <pre>Device(config-if)# exit</pre>	Exits interface configuration mode and enters global configuration mode.
Step 9	interface interface-type interface-number Example: <pre>Device(config)# interface GigabitEthernet0/0/0</pre>	Configures the LISP interface facing the site and enters interface configuration mode.
Step 10	description string Example: <pre>Device(config-if)# description Link To Site</pre>	Configures a description text for the interface.
Step 11	ip pim sparse-mode Example: <pre>Device(config-if)# ip pim sparse-mode</pre>	Enables Protocol Independent Multicast (PIM) on an interface for sparse-mode operation.

	Command or Action	Purpose
Step 12	end Example: Device(config-if)# end	Ends the current configuration session and returns to privileged EXEC mode.

Configuring LISP Multicast in VRFs

Perform this task to enable the LISP multicast functionality on an xTR with Virtual Routing and Forwarding (VRF) mode configured.

Before you begin

Ensure that generic multicast functionality has been enabled on the required devices of the LISP site and that PIM sparse mode has been enabled on the required interfaces of these devices.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vrf definition** *vrf-name*
4. **address-family** **ipv4**
5. **exit**
6. **exit**
7. **ip multicast-routing vrf** *vrf-name* [**distributed**]
8. Enter one of the following:
 - **ip pim vrf** *vrf-name* **rp-address** *ip-address*
 - **ip pim vrf** *vrf-name* **ssm** {**default** | **range** {*access-list-number* | *access-list-name*}}
9. **interface lisp** *interface-number*
10. **ip pim sparse-mode**
11. **exit**
12. **interface** *interface-type* *interface-number*
13. **vrf forwarding** *vrf-name*
14. **description** *string*
15. **ip pim** *sparse-mode*
16. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	vrf definition vrf-name Example: Device(config)# vrf definition VRF1	Configures a virtual routing and forwarding (VRF) routing table instance and enters VRF configuration mode.
Step 4	address-family ipv4 Example: Device(config-vrf)# address-family ipv4	Configures an address family for the VRF and enters VRF address family configuration mode.
Step 5	exit Example: Device(config-vrf-af)# exit	Exits VRF address family configuration mode and enters VRF configuration mode.
Step 6	exit Example: Device(config-vrf)# exit	Exits VRF configuration mode and enters global configuration mode.
Step 7	ip multicast-routing vrf vrf-name [distributed] Example: Device(config)# ip multicast-routing vrf VRF1 distributed	Enables IP multicast routing.
Step 8	Enter one of the following: <ul style="list-style-type: none"> • ip pim vrf vrf-name rp-address ip-address • ip pim vrf vrf-name ssm {default range {access-list-number access-list-name}} Example: Device(config)# ip pim vrf VRF1 rp-address 10.1.0.2 Example: Device(config)# ip pim vrf VRF1 ssm default	<ul style="list-style-type: none"> • Statically configures the address of a Protocol Independent Multicast (PIM) rendezvous point (RP) for multicast groups. • Defines the Source Specific Multicast (SSM) range of IP multicast addresses.
Step 9	interface lisp interface-number Example: Device(config)# interface lisp 22.10	Selects a LISP interface to configure and enters interface configuration mode.
Step 10	ip pim sparse-mode Example:	Enables Protocol Independent Multicast (PIM) on an interface for sparse-mode operation.

	Command or Action	Purpose
	<code>Device(config-if)# ip pim sparse-mode</code>	
Step 11	exit Example: <code>Device(config-if)# exit</code>	Exits interface configuration mode and enters global configuration mode.
Step 12	interface <i>interface-type interface-number</i> Example: <code>Device(config)# interface GigabitEthernet0/0/0</code>	Configures the LISP interface facing the site and enters interface configuration mode.
Step 13	vrf forwarding <i>vrf-name</i> Example: <code>Device(config-if)# vrf forwarding VRF1</code>	Enables VRF forwarding on the interface.
Step 14	description <i>string</i> Example: <code>Device(config-if)# description Link To Site</code>	Configures a description text for the interface.
Step 15	ip pim <i>sparse-mode</i> Example: <code>Device(config-if)# ip pim sparse-mode.</code>	Enables Protocol Independent Multicast (PIM) on an interface for sparse-mode operation.
Step 16	end Example: <code>Device(config-if)# end</code>	Ends the current configuration session and returns to privileged EXEC mode.

Verifying LISP Multicast

Perform this task to verify the configuration of LISP multicast routes on a device.

SUMMARY STEPS

1. **show ip mroute** *multicast-ip-address*
2. **ping** *multicast-ip-address*

DETAILED STEPS

Step 1 **show ip mroute** *multicast-ip-address*

Example:

The following example shows how the IP multicast routing table is displayed using the **show ip mroute** command:

```
Device# show ip mroute 239.4.4.4

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.4.4.4), 00:06:25/00:02:39, RP 10.1.0.2, flags: SJCL
  Incoming interface: Serial0/0, RPF nbr 10.1.0.2
  Outgoing interface list:
    Loopback2, Forward/Sparse, 00:06:24/00:02:39

(*, 224.0.1.40), 00:06:25/00:02:37, RP 10.1.0.2, flags: SJCL
  Incoming interface: Serial0/0, RPF nbr 10.1.0.2
  Outgoing interface list:
    Loopback2, Forward/Sparse, 00:06:24/00:02:37
```

Step 2 ping *multicast-ip-address***Example:**

The following example shows how to verify basic multicast network connectivity by pinging the multicast address:

```
Device# ping 239.4.4.4

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 239.4.4.4, timeout is 2 seconds:

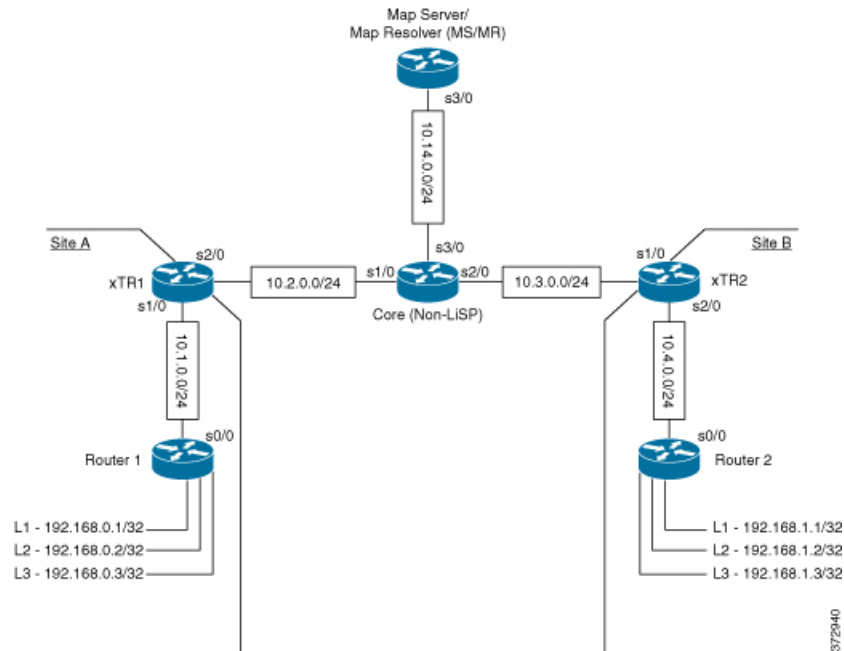
Reply to request 0 from 192.168.0.1, 15 ms
Reply to request 0 from 10.1.0.2, 58 ms
Reply to request 0 from 10.1.0.2, 58 ms
Reply to request 0 from 10.1.0.1, 35 ms
Reply to request 0 from 10.1.0.2, 34 ms
Reply to request 0 from 10.1.0.1, 15 ms
```


Configuration Examples for LISP Multicast

Example: Configuring LISP Multicast

The following example shows how to configure LISP Multicast in the topology given below:

Figure 20: LISP Multicast Topology



Router 1

The following example shows how to configure LISP multicast in Router 1:

Device# **show startup-config**

```
!
ip multicast-routing
!
interface Loopback1
 ip address 192.168.0.1 255.255.255.255
 ip pim sparse-mode
 ip igmp join-group 239.4.4.4
 serial restart-delay 0
!
interface Loopback2
 ip address 192.168.0.2 255.255.255.255
 ip pim sparse-mode
 ip igmp join-group 239.4.4.4
 serial restart-delay 0
!
interface Loopback3
 ip address 192.168.0.3 255.255.255.255
```

Example: Configuring LISP Multicast

```

ip pim sparse-mode
ip igmp join-group 239.4.4.4
serial restart-delay 0
!
interface Serial0/0
ip address 10.1.0.1 255.255.255.0
ip pim sparse-mode
serial restart-delay 0
!
router rip
version 2
network 10.0.0.0
network 192.168.0.0
default-information originate
!
ip forward-protocol nd
!
ip pim rp-address 10.1.0.2
!
!
End

```

The following example shows how to verify the configuration of LISP multicast routes in Router 1:

```
Device# show ip mroute
```

```

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.4.4.4), 00:00:49/00:02:16, RP 10.1.0.2, flags: SJCL
  Incoming interface: Serial0/0, RPF nbr 10.1.0.2
  Outgoing interface list:
    Loopback2, Forward/Sparse, 00:00:48/00:02:12

(*, 224.0.1.40), 00:00:49/00:02:11, RP 10.1.0.2, flags: SJCL
  Incoming interface: Serial0/0, RPF nbr 10.1.0.2
  Outgoing interface list:
    Loopback2, Forward/Sparse, 00:00:48/00:02:11

```

The following example shows how to verify basic multicast network connectivity from Router 1 by pinging the multicast address:

```
Device# ping 239.4.4.4
```

```
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 239.4.4.4, timeout is 2 seconds:

Reply to request 0 from 192.168.0.1, 9 ms
Reply to request 0 from 10.1.0.2, 48 ms
Reply to request 0 from 192.168.0.2, 16 ms
Reply to request 0 from 192.168.0.3, 16 ms
Reply to request 0 from 10.1.0.1, 38 ms
Reply to request 0 from 10.1.0.2, 38 ms
Reply to request 0 from 10.1.0.2, 29 ms
Reply to request 0 from 10.1.0.1, 9 ms
```

xTR1

The following example shows how to configure LISP multicast in xTR1:

```
Device# show startup-config

!
ip multicast-routing
!
interface LISP0
 ip pim sparse-mode
!
interface Serial1/0
 ip address 10.1.0.2 255.255.255.0
 ip pim sparse-mode
 serial restart-delay 0
!
interface Serial2/0
 ip address 10.2.0.1 255.255.255.0
 serial restart-delay 0
!
router lisp
 database-mapping 192.168.0.0/24 10.2.0.1 priority 1 weight 100
 ipv4 itr map-resolver 10.14.0.14
 ipv4 itr
 ipv4 etr map-server 10.14.0.14 key password123
 ipv4 etr
 exit
!
!
router rip
 version 2
 network 10.0.0.0
 default-information originate
!
ip pim rp-address 10.1.0.2
ip route 0.0.0.0 0.0.0.0 10.2.0.2
!
```

Router 2

The following example shows how to configure LISP multicast in Router 2:

```

Device# show startup-config

!
ip multicast-routing
!
interface Loopback1
 ip address 192.168.1.1 255.255.255.255
 ip pim sparse-mode
 ip igmp join-group 239.4.4.4
 serial restart-delay 0
!
interface Loopback2
 ip address 192.168.1.2 255.255.255.255
 ip pim sparse-mode
 ip igmp join-group 239.4.4.4
 serial restart-delay 0
!
interface Loopback3
 ip address 192.168.1.3 255.255.255.255
 ip pim sparse-mode
 ip igmp join-group 239.4.4.4
 serial restart-delay 0
!
interface Serial0/0
 ip address 10.4.0.2 255.255.255.0
 ip pim sparse-mode
 serial restart-delay 0
!
!
router rip
 version 2
 network 10.0.0.0
 network 192.168.1.0
 default-information originate
!
ip forward-protocol nd
!
!
ip pim rp-address 10.1.0.2
!
!
End

```

The following example shows how to verify the configuration of LISP multicast routes in Router 2:

```
Device# show ip mroute
```

```

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,
Y - Joined MDT-data group, y - Sending to MDT-data group,
G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
Q - Received BGP S-A Route, q - Sent BGP S-A Route,
V - RD & Vector, v - Vector, p - PIM Joins on route,
x - VxLAN group

```

```

Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.4.4.4), 00:12:59/00:02:01, RP 10.4.0.1, flags: SJCL
  Incoming interface: Serial0/0, RPF nbr 10.4.0.1
  Outgoing interface list:
    Loopback2, Forward/Sparse, 00:12:58/00:02:01

(*, 224.0.1.40), 00:12:59/00:02:03, RP 10.4.0.1, flags: SJCL
  Incoming interface: Serial0/0, RPF nbr 10.4.0.1
  Outgoing interface list:
    Loopback2, Forward/Sparse, 00:12:58/00:02:03

```

The following example shows how to verify basic multicast network connectivity from Router 2 by pinging the multicast address:

```

Device# ping 239.4.4.4

Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 239.4.4.4, timeout is 2 seconds:

Reply to request 0 from 192.168.1.1, 2 ms
Reply to request 0 from 10.3.0.2, 26 ms
Reply to request 0 from 10.4.0.1, 26 ms
Reply to request 0 from 192.168.1.2, 2 ms
Reply to request 0 from 192.168.1.3, 8 ms
Reply to request 0 from 10.4.0.1, 16 ms
Reply to request 0 from 10.4.0.1, 16 ms
Reply to request 0 from 10.4.0.2, 2 ms

```

xTR2

The following example shows how to configure LISP multicast in xTR2:

```

Device# show startup-config

!
ip multicast-routing
!
interface LISP0
 ip pim sparse-mode
!
!
interface Serial1/0
 ip address 10.3.0.2 255.255.255.0
 serial restart-delay 0
!
interface Serial2/0
 ip address 10.4.0.1 255.255.255.0
 ip pim sparse-mode
 serial restart-delay 0
!
!
router lisp
 database-mapping 192.168.1.0/24 10.3.0.2 priority 1 weight 100

```

```

ipv4 itr map-resolver 10.14.0.14
ipv4 itr
ipv4 etr map-server 10.14.0.14 key Amel
ipv4 etr
exit
!
router rip
version 2
network 10.0.0.0
default-information originate
!
ip pim rp-address 10.1.0.2
ip route 0.0.0.0 0.0.0.0 10.3.0.1
!

```

MS/MR

The following example shows how to configure LISP multicast in MS/MR:

```
Device# show startup-config
```

```

!
ip multicast-routing
!
interface Serial3/0
ip address 10.14.0.14 255.255.255.0
serial restart-delay 0
!
!
router lisp
site Site-A
authentication-key password123
eid-prefix 192.168.0.0/24
exit
!
site Site-B
authentication-key Amel
eid-prefix 192.168.1.0/24
exit
!
ipv4 map-server
ipv4 map-resolver
exit
!
ip route 0.0.0.0 0.0.0.0 10.14.0.1
!

```

Core

The following example shows how to configure LISP multicast in the Core router:

```
Device# show startup-config
```

```

!
ip multicast-routing

```

```

!
interface Ethernet0/0
 ip address 10.14.0.1 255.255.255.0
 serial restart-delay 0
!
interface Serial1/0
 ip address 10.2.0.2 255.255.255.0
 serial restart-delay 0
!
interface Serial2/0
 ip address 10.3.0.1 255.255.255.0
 serial restart-delay 0
!

```

Example: Configuring LISP Multicast in VRFs

The following example shows how to enable and configure a simple LISP site with one IPv4 Routing locator (RLOC) and one IPv4 Endpoint identifier (EID) using xTR, a device which functions both as an Ingress tunnel router (ITR) and an Egress tunnel router (ETR), functionality and using a LISP map server and map resolver for mapping services:

```

Device> enable
Device# configure terminal
Device(config)# vrf definition VRF1
Device(config-vrf)# address-family ipv4
Device(config-vrf-af)# exit
Device(config-vrf)# exit
Device(config)# ip multicast-routing vrf VRF1 [distributed]
Device(config)# ip pim vrf VRF1 ssm range LIST1
Device(config)# router lisp 22
Device(config-router-lisp)# eid-table vrf VRF1 instance-id 10
Device(config-router-lisp-eid-table)# database-mapping 198.51.100.0/24 192.0.2.10 priority
 1 weight 100
Device(config-router-lisp-eid-table)# exit
Device(config-router-lisp)# ipv4 itr
Device(config-router-lisp)# ipv4 etr
Device(config-router-lisp)# ipv4 itr map-resolver 192.0.2.10
Device(config-router-lisp)# ipv4 etr map-server 192.0.2.10 key 0 some-key
Device(config-router-lisp)# exit
Device(config)# interface lisp 22.10
Device(config-if)# ip pim sparse-mode
Device(config-if)# exit
Device(config)# ip route 0.0.0.0 0.0.0.0 192.0.2.20
Device(config)# end

```

Additional References for LISP Multicast

The following sections provide references related to the Locator ID Separation Protocol.

Related Documents

Document Title	Location
Cisco IOS commands	Cisco IOS Master Command List, All Releases
LISP commands	Cisco IOS IP Routing: LISP Command Reference

Standards

Standard	Title
Address family numbers	IANA Address Family Numbers

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 6830	Locator/ID Separation Protocol (LISP) http://tools.ietf.org/html/
RFC 6831	LISP Multicast http://tools.ietf.org/html/rfc6831
RFC 6832	Interworking LISP and Non-LISP Sites http://tools.ietf.org/html/rfc6832
RFC 6833	LISP Map Server Interface http://tools.ietf.org/html/rfc6833
RFC 6834	LISP Map-Versioning http://tools.ietf.org/html/rfc6834
RFC 6835	LISP Internet Groper http://tools.ietf.org/html/rfc6835
RFC 6836	LISP Alternative Topology (LISP+ALT) http://tools.ietf.org/html/rfc6836

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature Information for LISP Multicast

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 3: Feature Information for LISP Multicast

Feature Name	Releases	Feature Information
LISP Multicast		<p>The LISP Multicast feature introduces support for carrying multicast traffic over a Locator ID Separation Protocol (LISP) overlay and allows source multicast sites and receiver multicast sites to send and receive multicast packets over a unicast RLOC core.</p>



CHAPTER 5

LISP Shared Model Virtualization

This guide describes how to configure Locator ID Separation Protocol (LISP) shared model virtualization using Software on all LISP-related devices, including the Egress Tunnel Router, Ingress Tunnel Router (ITR), Proxy ETR (PETR), Proxy ITR (PITR), Map Resolver (MR), and Map Server (MS).

LISP implements a new routing architecture that utilizes a "level of indirection" to separate an IP address into two namespaces: Endpoint Identifiers (EIDs), which are assigned to end-hosts, and Routing Locators (RLOCs), which are assigned to devices (primarily routers) that make up the global routing system. Splitting EID and RLOC functions yields several advantages including: improved routing system scalability, multihoming with ingress traffic engineering; efficient IPv6 Transition support; high-scale virtualization/multitenancy support; data center/VM-mobility support, including session persistence across mobility events; and seamless mobile node support.

- [Finding Feature Information, on page 105](#)
- [Information About LISP Shared Model Virtualization, on page 105](#)
- [How to Configure LISP Shared Model Virtualization, on page 110](#)
- [Configuration Examples for LISP Shared Model Virtualization, on page 142](#)
- [Additional References, on page 142](#)
- [Feature Information for LISP Shared Model Virtualization, on page 144](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About LISP Shared Model Virtualization

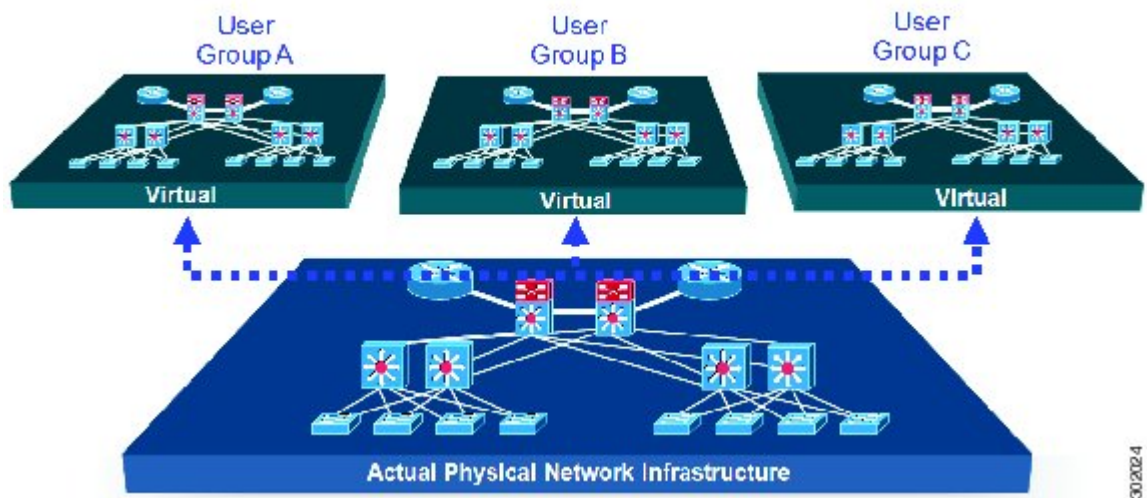
Overview of LISP Virtualization

Deploying physical network infrastructure requires both capital investments for hardware, as well as manpower investments for installation and operational management support. When distinct user groups within an

organization desire to control their own networks, it rarely makes economic sense for these user groups to deploy and manage separate physical networks. Physical plants are rarely utilized to their fullest, resulting in stranded capacity (bandwidth, processor, memory, etc.). In addition, the power, rack space, and cooling needs to physical plants do not satisfy modern “green” requirements. Network virtualization offers the opportunity to satisfy organizational needs, while efficiently utilizing physical assets.

The purpose of network virtualization, as shown in the figure below, is to create multiple, logically separated topologies across one common physical infrastructure.

Figure 21: LISP Deployment Environment



When considering the deployment of a virtualized network environment, take into account both the device and the path level.

Device Level Virtualization

Virtualization at the device level entails the use of the virtual routing and forwarding (VRF) to create multiple instances of Layer 3 routing tables, as illustrated in the figure below. VRFs provide segmentation across IP addresses, allowing for overlapped address space and traffic separation. Separate routing, QoS, security, and management policies can be applied to each VRF instance. An IGP or EGP routing process is typically enabled within a VRF, just as it would be in the global (default) routing table. As described in detail below, LISP binds VRFs to instance IDs for similar purposes.

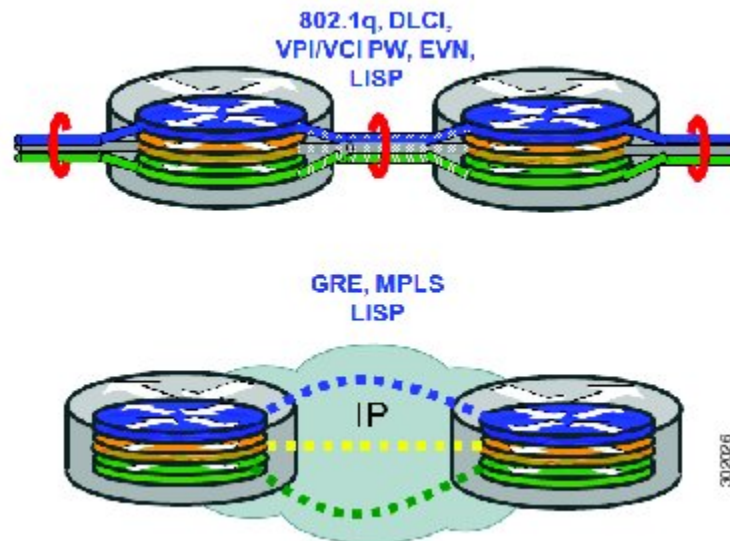
Figure 22: Device Level Virtualization



Path Level Virtualization

VRF table separation is maintained across network paths using any number of traditional mechanisms, as illustrated in the figure below. Single-hop path segmentation (hop-by-hop) is typically accomplished by techniques such as 802.1q VLANs, VPI/VCI PW, or EVN. LISP can also be used. Traditional multi-hop mechanisms include MPLS and GRE tunnels. As described in detail below, LISP binds VRFs to instance IDs (IIDs), and then these IIDs are included in the LISP header to provide data plane (traffic flow) separation for single or multihop needs.

Figure 23: Path Level Virtualization



LISP Virtualization at the Device Level

Recalling that LISP implements Locator ID separation and, in so doing, creates two namespaces (EIDs and RLOCs), it is easy to see that LISP virtualization can consider both EID and RLOC namespaces for virtualization. That is, either or both can be virtualized.

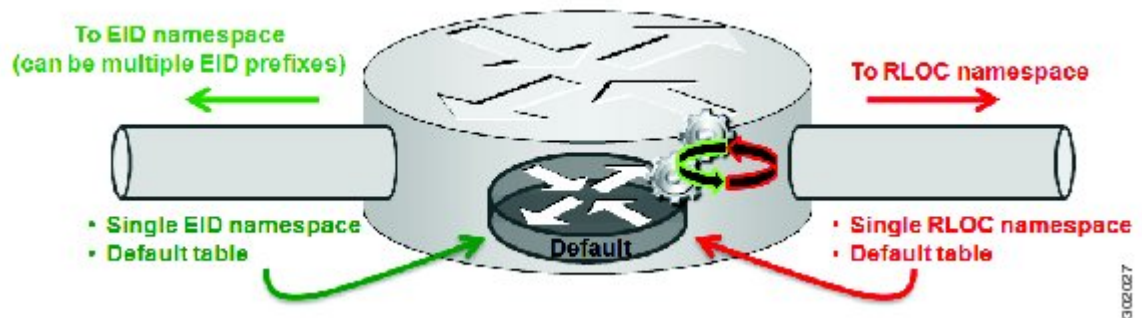
- EID virtualization—Enabled by binding a LISP instance ID to an EID VRF. Instance IDs are numerical tags defined in the LISP canonical address format (LCAF) draft, and are used to maintain address space segmentation in both the control plane and data plane.
- RLOC virtualization—Tying locator addresses and associated mapping services to the specific VRF within which they are reachable enables RLOC virtualization.

Because LISP considers virtualization of both EID and RLOC namespaces, two models of operation are defined: shared model and parallel model. For completeness, the discussions below begin first with a review of the default (non-virtualized) model of LISP, and then cover the details of shared and parallel models.

Default (Non-Virtualized) LISP Model

By default, LISP is not virtualized in either EID space or RLOC space. That is, unless otherwise configured, both EID and RLOC addresses are resolved in the default (global) routing table. This concept is illustrated in the figure below.

Figure 24: Default (Non-Virtualized) LISP Model (Resolves Both EID and RLOC Addresses in the Default (Global) Routing Table).

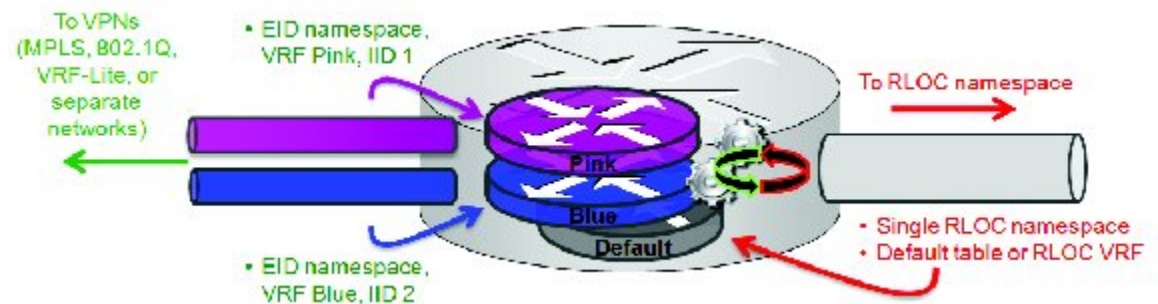


As shown in the figure above, both EID and RLOC addresses are resolved in the default table. The mapping system must also be reachable via the default table. This default model can be thought of as a single instantiation of the parallel model of LISP virtualization where EID and RLOC addresses are within the same namespace such as is the case in this default table.

LISP Shared Model Virtualization

LISP shared model virtualized EID space is created by binding VRFs associated with an EID space to Instance IDs. A common, shared locator space is used by all virtualized EIDs. This concept is illustrated in the figure below.

Figure 25: LISP shared model virtualization resolves EIDs within VRFs tied to Instance IDs. RLOC addresses are resolved in a common (shared) address space. The default (global) routing table is shown as the shared space.

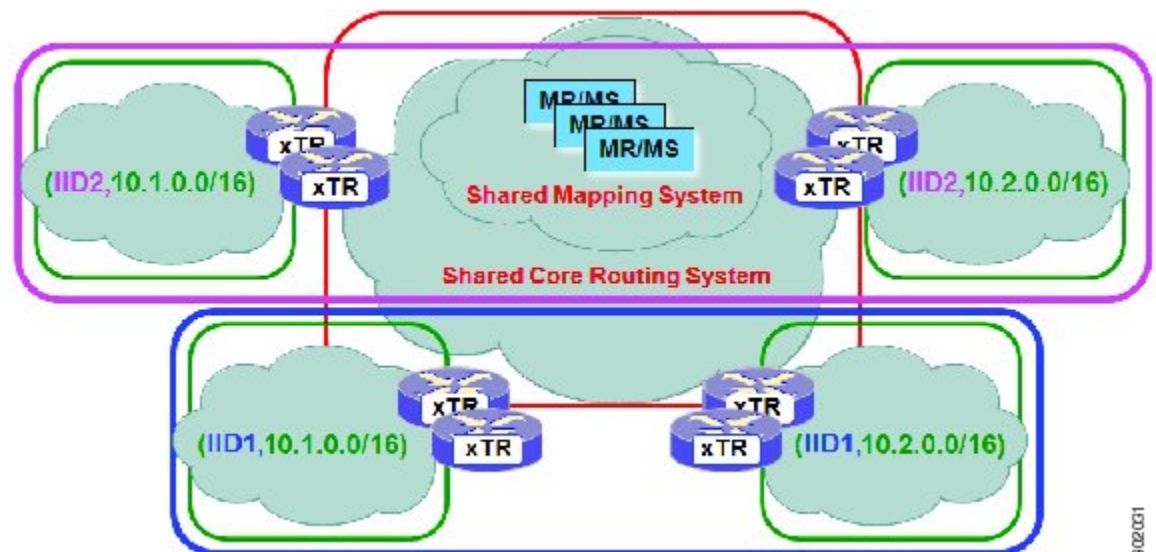


As shown in the figure above, EID space is virtualized through its association with VRFs, and these VRFs are tied to LISP Instance IDs to segment the control plane and data plane in LISP. A common, shared locator space, the default (global) table as shown in the figure above, is used to resolve RLOC addresses for all virtualized EIDs. The mapping system must also be reachable via the common locator space.

LISP Shared Model Virtualization Architecture

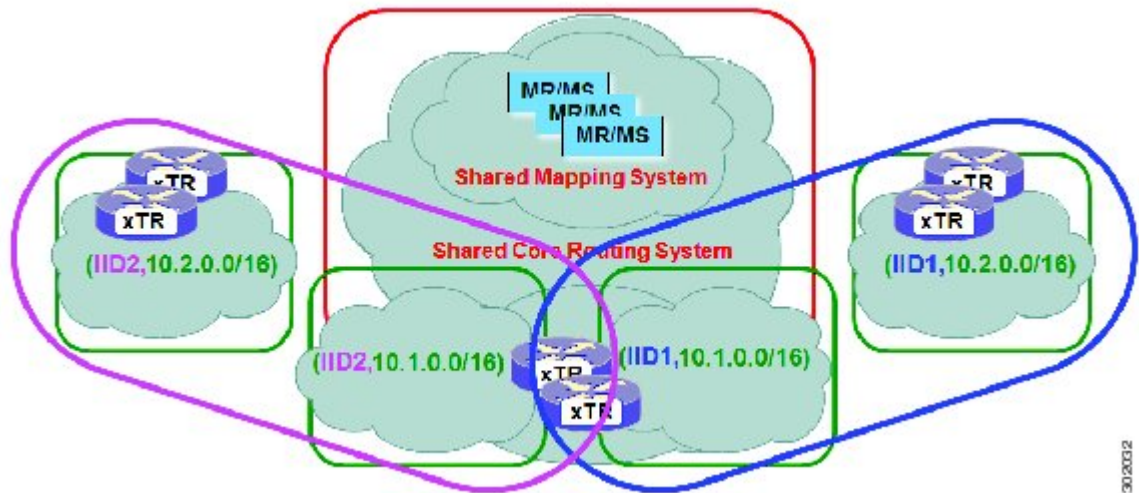
Architecturally, LISP shared model virtualization can be deployed in single or multitenancy configurations. In the shared model single tenancy case, xTRs are dedicated to a customer but share infrastructure with other customers. Each customer and all sites associated with it use the same instance ID and are part of a VPN using their own EID namespace as shown in the figure below.

Figure 26: In a LISP shared model single tenancy use case, customers use their own xTRs and a shared common core network and mapping system. LISP instance IDs segment the LISP data plane and control plane.



In the shared model multitenancy case, a set of xTRs is shared (virtualized) among multiple customers. These customers also share a common infrastructure with other single and multitenant customers. Each customer and all sites associated with it use the same instance ID and are part of a VPN using their own EID namespace as shown in the figure below.

Figure 27: In a LISP shared model multitenancy use case, customer's use shared xTRs and a shared common core network and mapping system. LISP instance IDs segment the LISP data plane and control plane.



LISP Shared Model Virtualization Implementation Considerations and Caveats

When LISP Shared Model is implemented, several important considerations and caveats are important. Instance IDs must be unique to an EID VRF. Review the example below:

```
xTR-1 (config) # vrf definition alpha
xTR-1 (config-vrf) # address-family ipv4
xTR-1 (config-vrf-af) # exit
xTR-1 (config) # vrf definition beta
xTR-1 (config-vrf) # address-family ipv4
xTR-1 (config-vrf-af) # exit
xTR-1 (config-vrf) # exit
xTR-1 (config) # router lisp
xTR-1 (config-router-lisp) # eid-table vrf alpha instance-id 101
xTR-1 (config-router-lisp-eid-table) # exit
xTR-1 (config-router-lisp) # eid-table vrf beta instance-id 101
Instance ID 101 is bound to the vrf alpha EID table.
```

In the above example, two EID VRFs are created: alpha and beta. Under the **router lisp** command, an EID table VRF named alpha is specified and associated with the instance ID 101. Next, an EID table VRF named beta is specified and also associated with the instance ID 101. As indicated by the router, this is not permissible since instance ID 101 is already associated with the EID VRF named alpha. That is, you cannot connect the same instance-id to more than one EID VRF.

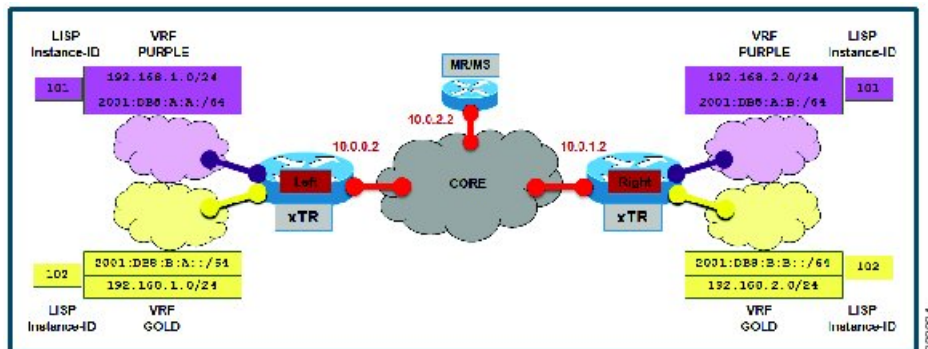
How to Configure LISP Shared Model Virtualization

Configure Simple LISP Shared Model Virtualization

Perform this task to enable and configure LISP ITR/ETR (xTR) functionality with LISP map server and map resolver to implement LISP shared model virtualization. This LISP shared model reference configuration is for a very simple two-site LISP topology, including xTRs and an MS/MR.

The configuration implemented in this task and illustrated in the figure below shows a basic LISP shared model virtualization solution. In this example, two LISP sites are deployed, each containing two VRFs: PURPLE and GOLD. LISP is used to provide virtualized connectivity between these two sites across a common IPv4 core, while maintaining address separation between the two VRFs.

Figure 28: Simple LISP Site with virtualized IPv4 and IPv6 EIDs and a shared IPv4 core



Each LISP Site uses a single edge router configured as both an ITR and ETR (xTR), with a single connection to its upstream provider. The RLOC is IPv4, and IPv4 and IPv6 EID prefixes are configured. Each LISP site registers to a map server/map resolver (MS/MR) device located in the network core within the shared RLOC address space. The topology used in this most basic LISP configuration is shown in the figure above.

The components illustrated in the topology shown in the figure above are described below:

- **LISP site:**

- The CPE functions as a LISP ITR and ETR (xTR).
- Both LISP xTRs have two VRFs: GOLD and PURPLE, with each VRF containing both IPv4 and IPv6 EID-prefixes, as shown in the figure above. Note the overlapping prefixes, used for illustration purposes. A LISP instance-id is used to maintain separation between two VRFs. Note that in this example, the share key is configured "per-site" and not "per-VRF." (Case 2 illustrates a configuration where the shared key is per-VPN.)
- Each LISP xTR has a single RLOC connection to a shared IPv4 core network.

- **Mapping system:**

- One map server/map resolver system is shown in the figure above and assumed available for the LISP xTR to register to. The MS/MR has an IPv4 RLOC address of 10.0.2.2, within the shared IPv4 core.
- The map server site configurations are virtualized using LISP instance-ids to maintain separation between the two VRFs.

Perform the steps in this task (once through for each xTR in the LISP site) to enable and configure LISP ITR and ETR (xTR) functionality when using a LISP map-server and map-resolver for mapping services. The example configurations at the end of this task show the full configuration for two xTRs (xTR1 and xTR2).

Before you begin

The configuration below assumes that the referenced VRFs were created using the **vrf definition** command.

SUMMARY STEPS

1. **configure terminal**
2. **router lisp**
3. **eid-table vrfvrf-name instance-id instance-id**
4. Do one of the following:
 - **database-mapping EID-prefix/prefix-length locator priority priority weight weight**
 - **database-mapping EID-prefix/prefix-length locator priority priority weight weight**
5. Repeat Step 4 until all EID-to-RLOC mappings for the LISP site are configured.
6. **exit**
7. **ipv4 itr**
8. **ipv4 etr**
9. **ipv4 itr map-resolver map-resolver-address**
10. **ipv4 etr map-server map-server-address key key-type authentication-key**
11. **ipv6 itr**
12. **ipv6 etr**
13. **ipv6 itr map-resolver map-resolver-address**
14. **ipv6 etr map-server map-server-address key key-type authentication-key**
15. **exit**
16. **ip route ipv4-prefix next-hop**
17. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 2	router lisp Example: Router(config)# router lisp	Enters LISP configuration mode (software only).
Step 3	eid-table vrfvrf-name instance-id instance-id Example: Router(config-router-lisp)# eid-table vrf GOLD instance-id 102	Configures an association between a VRF table and a LISP instance ID, and enters eid-table configuration submode. <ul style="list-style-type: none"> • In this example, the VRF table GOLD and instance-id 102 are associated together.
Step 4	Do one of the following: <ul style="list-style-type: none"> • database-mapping EID-prefix/prefix-length locator priority priority weight weight • database-mapping EID-prefix/prefix-length locator priority priority weight weight 	Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site. <ul style="list-style-type: none"> • In the first example, a single IPv4 EID prefix, 192.168.1.0/24, is being associated with the single IPv4 RLOC 10.0.0.2.

	Command or Action	Purpose
	<p>Example:</p> <pre>Router(config-router-lisp-eid-table)# database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 100</pre> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# database-mapping 192.168.1.0/24 ipv4-interface Ethernet0/0 priority 1 weight 100</pre>	<ul style="list-style-type: none"> In the second example, the alternative configuration shows the use of the dynamic interface form of the database-mapping command. This form is useful when the RLOC address is obtained dynamically, such as via DHCP.
Step 5	<p>Repeat Step 4 until all EID-to-RLOC mappings for the LISP site are configured.</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# database-mapping 2001:db8:b:a::/64 10.0.0.2 priority 1 weight 100</pre>	Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site.
Step 6	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# exit</pre>	Exits eid-table configuration submode and returns to LISP configuration mode.
Step 7	<p>ipv4 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr</pre>	Enables LISP ITR functionality for the IPv4 address family.
Step 8	<p>ipv4 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr</pre>	Enables LISP ETR functionality for the IPv4 address family.
Step 9	<p>ipv4 itr map-resolver map-resolver-address</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 10.0.2.2</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable using its IPv4 locator address. (See the <i>LISP Command Reference Guide</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>

	Command or Action	Purpose
Step 10	<p>ipv4 etr map-server <i>map-server-address key key-type authentication-key</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr map-server 10.0.2.2 key 0 Left-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key for which this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> The map server must be configured with EID prefixes and instance IDs matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map-server is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 11	<p>ipv6 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr</pre>	<p>Enables LISP ITR functionality for the IPv6 address family.</p>
Step 12	<p>ipv6 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 etr</pre>	<p>Enables LISP ETR functionality for the IPv6 address family.</p>
Step 13	<p>ipv6 itr map-resolver <i>map-resolver-address</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 itr map-resolver 10.0.2.2</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv6 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map-resolver is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 14	<p>ipv6 etr map-server <i>map-server-address key key-type authentication-key</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 etr map-server 10.0.2.2 key 0 Left-key</pre>	<p>Configures a locator address for the LISP map-server and an authentication key that this router, acting as an IPv6 LISP ETR, will use to register to the LISP mapping system.</p> <ul style="list-style-type: none"> The map-server must be configured with EID prefixes and instance IDs matching those configured on this ETR and with an identical authentication key.

	Command or Action	Purpose
		<p>Note The locator address of the map-server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map-server is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 15	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 16	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 10.0.0.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> LISP-encapsulated to a LISP site when traffic is LISP-to-LISP natively forwarded when traffic is LISP-to-non-LISP Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP EID and the destination matches one of the following entries: <ul style="list-style-type: none"> a current map-cache entry a default route with a legitimate next-hop no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
Step 17	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	Exits global configuration mode.

Example:

The examples below show the complete configuration for the LISP topology illustrated in the figure shown above the task steps and follows the examples in the steps in this task. On the xTRs, the VRFs and EID prefixes are assumed to be attached to VLANs configured on the devices.

Example configuration for the Left xTR:

```
hostname Left-xTR
```

```

!
ipv6 unicast-routing
!
vrf definition PURPLE
  address-family ipv4
  exit
  address-family ipv6
  exit
!
vrf definition GOLD
  address-family ipv4
  exit
  address-family ipv6
  exit
!
interface Ethernet0/0
  ip address 10.0.0.2 255.255.255.0
!
interface Ethernet1/0.1
  encapsulation dot1q 101
  vrf forwarding PURPLE
  ip address 192.168.1.1 255.255.255.0
  ipv6 address 2001:DB8:A:A::1/64
!
interface Ethernet1/0.2
  encapsulation dot1q 102
  vrf forwarding GOLD
  ip address 192.168.1.1 255.255.255.0
  ipv6 address 2001:DB8:B:A::1/64
!
router lisp
  eid-table vrf PURPLE instance-id 101
    database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
    database-mapping 2001:DB8:A:A::/64 10.0.0.2 priority 1 weight 1
  eid-table vrf GOLD instance-id 102
    database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
    database-mapping 2001:DB8:B:A::/64 10.0.0.2 priority 1 weight 1
  exit
!
  ipv4 itr map-resolver 10.0.2.2
  ipv4 itr
  ipv4 etr map-server 10.0.2.2 key Left-key
  ipv4 etr
  ipv6 itr map-resolver 10.0.2.2
  ipv6 itr
  ipv6 etr map-server 10.0.2.2 key Left-key
  ipv6 etr
  exit
!
ip route 0.0.0.0 0.0.0.0 10.0.0.1
!

```

Example configuration for Right xTR:

```

hostname Right-xTR
!
ipv6 unicast-routing
!
vrf definition PURPLE
  address-family ipv4
  exit
  address-family ipv6
  exit
!

```

```

vrf definition GOLD
  address-family ipv4
  exit
  address-family ipv6
  exit
!
interface Ethernet0/0
  ip address 10.0.1.2 255.255.255.0
!
interface Ethernet1/0.1
  encapsulation dot1q 101
  vrf forwarding PURPLE
  ip address 192.168.2.1 255.255.255.0
  ipv6 address 2001:DB8:A:B::1/64
!
interface Ethernet1/0.2
  encapsulation dot1q 102
  vrf forwarding GOLD
  ip address 192.168.2.1 255.255.255.0
  ipv6 address 2001:DB8:B:B::1/64
!
router lisp
  eid-table vrf PURPLE instance-id 101
    database-mapping 192.168.2.0/24 10.0.1.2 priority 1 weight 1
    database-mapping 2001:DB8:A:B::/64 10.0.1.2 priority 1 weight 1
  eid-table vrf GOLD instance-id 102
    database-mapping 192.168.2.0/24 10.0.1.2 priority 1 weight 1
    database-mapping 2001:DB8:B:B::/64 10.0.1.2 priority 1 weight 1
  exit
!
  ipv4 itr map-resolver 10.0.2.2
  ipv4 itr
  ipv4 etr map-server 10.0.2.2 key Right-key
  ipv4 etr
  ipv6 itr map-resolver 10.0.2.2
  ipv6 itr
  ipv6 etr map-server 10.0.2.2 key Right-key
  ipv6 etr
  exit
!
ip route 0.0.0.0 0.0.0.0 10.0.1.1
!

```

Configuring a Private LISP Mapping System for LISP Shared Model Virtualization

Perform this task to configure and enable standalone LISP map server/map resolver functionality for LISP shared model virtualization. In this task, a Cisco router is configured as a standalone map server/map resolver (MR/MS) for a private LISP mapping system. Because the MR/MS is configured as a stand-alone device, it has no need for LISP Alternate Logical Topology (ALT) connectivity. All relevant LISP sites must be configured to register with this map server so that this map server has full knowledge of all registered EID Prefixes within the (assumed) private LISP system.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router lisp**
4. **site *site-name***
5. **authentication-key [*key-type*] *authentication-key***

6. **eid-prefix instance-id** *instance-id EID-prefix*
7. **eid-prefix instance-id** *instance-id EID-prefix*
8. **exit**
9. **ipv4 map-resolver**
10. **ipv4 map-server**
11. **ipv6 map-resolver**
12. **ipv6 map-server**
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	router lisp Example: Router(config)# router lisp	Enters LISP configuration mode (IOS only).
Step 4	site site-name Example: Router(config-router-lisp)# site Left	Specifies a LISP site named Left and enters LISP site configuration mode. <p>Note A LISP site name is locally significant to the map server on which it is configured. It has no relevance anywhere else. This name is used solely as an administrative means of associating EID-prefix or prefixes with an authentication key and other site-related mechanisms.</p>
Step 5	authentication-key [key-type] authentication-key Example: Router(config-router-lisp-site)# authentication-key 0 Left-key	Configures the password used to create the SHA-2 HMAC hash for authenticating the map register messages sent by an ETR when registering to the map server. <p>Note The LISP ETR must be configured with an identical authentication key as well as matching EID prefixes and instance IDs.</p>
Step 6	eid-prefix instance-id instance-id EID-prefix Example:	Configures an EID prefix and instance ID that are allowed in a map register message sent by an ETR when registering to this map server. Repeat this step as necessary to configure additional EID prefixes under this LISP site.

	Command or Action	Purpose
	<pre>Router(config-router-lisp-site)# eid-prefix instance-id 102 192.168.1.0/24</pre>	<ul style="list-style-type: none"> In this example, the IPv4 EID prefix 192.168.1.0/24 and instance ID 102 are associated together. To complete this task, an IPv6 EID prefix is required.
Step 7	<p>eid-prefix instance-id instance-id EID-prefix</p> <p>Example:</p> <pre>Router(config-router-lisp-site)# eid-prefix instance-id 102 2001:db8:a:b::/64</pre>	<p>Configures an EID prefix and instance ID that are allowed in a map register message sent by an ETR when registering to this map server.</p> <ul style="list-style-type: none"> In this example, the IPv6 EID prefix 2001:db8:a:b::/64 and instance ID 102 are associated together.
Step 8	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp-site)# exit</pre>	Exits LISP site configuration mode and returns to LISP configuration mode.
Step 9	<p>ipv4 map-resolver</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 map-resolver</pre>	Enables LISP map resolver functionality for EIDs in the IPv4 address family.
Step 10	<p>ipv4 map-server</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 map-server</pre>	Enables LISP map server functionality for EIDs in the IPv4 address family.
Step 11	<p>ipv6 map-resolver</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 map-resolver</pre>	Enables LISP map resolver functionality for EIDs in the IPv6 address family.
Step 12	<p>ipv6 map-server</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv6 map-server</pre>	Enables LISP map server functionality for EIDs in the IPv6 address family.
Step 13	<p>end</p> <p>Example:</p> <pre>Router(config-router-lisp)# end</pre>	Exits LISP configuration mode and returns to privileged EXEC mode.

Example:

Example configuration for the map server/map resolver.

```
hostname MSMR
```

```

!
interface Ethernet0/0
 ip address 10.0.2.2 255.255.255.0
!
router lisp
!
site Left
 authentication-key Left-key
 eid-prefix instance-id 101 192.168.1.0/24
 eid-prefix instance-id 101 2001:DB8:A:A::/64
 eid-prefix instance-id 102 192.168.1.0/24
 eid-prefix instance-id 102 2001:DB8:B:A::/64
 exit
!
site Right
 authentication-key Right-key
 eid-prefix instance-id 101 192.168.2.0/24
 eid-prefix instance-id 101 2001:DB8:A:B::/64
 eid-prefix instance-id 102 192.168.2.0/24
 eid-prefix instance-id 102 2001:DB8:B:B::/64
 exit
!
ipv4 map-server
ipv4 map-resolver
ipv6 map-server
ipv6 map-resolver
exit
!
ip route 0.0.0.0 0.0.0.0 10.0.2.1

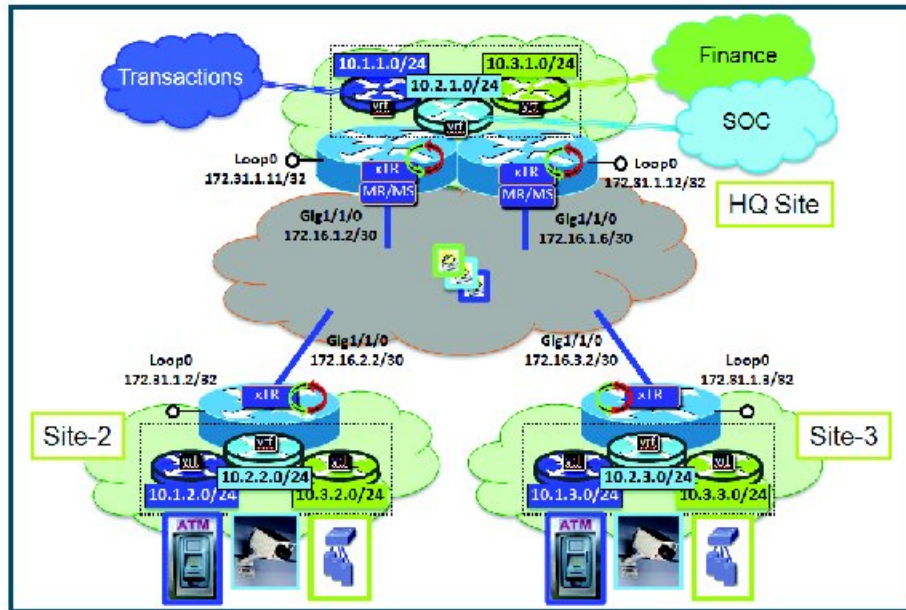
```

Configure Large-Scale LISP Shared Model Virtualization

Perform this task to enable and configure LISP ITR/ETR (xTR) functionality with LISP map server and map resolver to implement LISP shared model virtualization. This LISP shared model reference configuration is for a large-scale, multiple-site LISP topology, including xTRs and multiple MS/MRs.

The configuration demonstrated in this task shows a more complex, larger scale LISP virtualization solution. In this task, an enterprise is deploying LISP Shared Model where EID space is virtualized over a shared, common core network. A subset of their entire network is illustrated in Figure 12. In this figure, three sites are shown: a multihomed "Headquarters" (HQ) site, and two remote office sites. The HQ site routers are deployed as xTRs and also as map resolver/map servers. The remote site routers only act as xTRs, and use the MS/MRs at the HQ site for LISP control plane support.

Figure 29: Large Scale LISP Site with Virtualized IPv4 EIDs and a Shared IPv4 Core



The components illustrated in the topology shown in the figure above are described below:

- **LISP site:**

- Each CPE router functions as a LISP ITR and ETR (xTR), as well as a Map-Server/Map-Resolver (MS/MR).
- Both LISP xTRs have three VRFs: TRANS (for transactions), SOC (for security operations), and FIN (for financials). Each VRF contains only IPv4 EID-prefixes. Note that no overlapping prefixes are used, but segmentation between each VRF by LISP instance-ids makes this possible. Also note that in this example, the separate authentication key is configured “per-vrf” and not “per-site.” This affects both the xTR and MS configurations.
- The HQ LISP Site is multi-homed to the shared IPv4 core, but each xTR at the HQ site has a single RLOC.
- Each CPE also functions as an MS/MR to which the HQ and Remote LISP sites can register.
- The map server site configurations are virtualized using LISP instance IDs to maintain separation between the three VRFs.

- **LISP remote sites:**

- Each remote site CPE router functions as a LISP ITR and ETR (xTR).
- Each LISP xTRs has the same three VRFs as the HQ Site: TRANS, SOC, and FIN. Each VRF contains only IPv4 EID-prefixes.
- Each remote site LISP xTR has a single RLOC connection to a shared IPv4 core network.

Before you begin

The configuration below assumes that the referenced VRFs were created using the **vrf definition** command.

SUMMARY STEPS

1. **configure terminal**
2. **router lisp**
3. **site *site-name***
4. **authentication-key [*key-type*] *authentication-key***
5. **eid-prefix *instance-id* *instance-id* *EID-prefix/prefix-length* **accept-more-specifics****
6. **exit**
7. Repeat steps 3 through 6 for each LISP site to be configured.
8. **ipv4 map-resolver**
9. **ipv4 map-server**
10. **eid-table *vrfvrf-name* *instance-id* *instance-id***
11. **database-mapping *EID-prefix/prefix-length* *locator* **priority** *priority* **weight** *weight***
12. Repeat Step 11 until all EID-to-RLOC mappings within this eid-table vrf and instance ID for the LISP site are configured.
13. **ipv4 etr map-server *map-server-address* **key** *key-type* *authentication-key***
14. Repeat Step 13 to configure another locator address for the same LISP map server
15. **exit**
16. **ipv4 itr map-resolver *map-resolver-address***
17. Repeat Step 16 to configure another locator address for the LISP map resolver
18. **ipv4 itr**
19. **ipv4 etr**
20. **exit**
21. **ip route *ipv4-prefix* *next-hop***
22. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 2	router lisp Example: Router(config)# router lisp	Enters LISP configuration mode (software only).
Step 3	site <i>site-name</i> Example:	Specifies a LISP site named TRANS and enters LISP site configuration mode.

	Command or Action	Purpose
	<pre>Router(config-router-lisp)# site TRANS</pre>	<p>Note A LISP site name is locally significant to the map server on which it is configured. It has no relevance anywhere else. This name is used solely as an administrative means of associating EID-prefix or prefixes with an authentication key and other site-related mechanisms.</p>
Step 4	<p>authentication-key <i>[key-type] authentication-key</i></p> <p>Example:</p> <pre>Router(config-router-lisp-site)# authentication-key 0 TRANS-key</pre>	<p>Configures the password used to create the SHA-2 HMAC hash for authenticating the map register messages sent by an ETR when registering to the map server.</p> <p>Note The LISP ETR must be configured with an identical authentication key as well as matching EID prefixes and instance IDs.</p>
Step 5	<p>eid-prefix instance-id instance-id EID-prefix/prefix-length accept-more-specifics</p> <p>Example:</p> <pre>Router(config-router-lisp-site)# eid-prefix instance-id 1 10.1.0.0/16 accept-more-specifics</pre>	<p>Configures an EID prefix and instance ID that are allowed in a map register message sent by an ETR when registering to this map server. Repeat this step as necessary to configure additional EID prefixes under this LISP site.</p> <ul style="list-style-type: none"> In the example, EID-prefix 10.1.0.0/16 and instance-id 1 are associated together. The EID-prefix 10.1.0.0/16 is assumed to be an aggregate covering all TRANS EID-prefixes at all LISP Sites. The keyword accept-more-specifics is needed in this case to allow each site to register its more-specific EID-prefix contained within that aggregate. If aggregation is not possible, simply enter all EID-prefixes integrated within instance-id 1.
Step 6	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp-site)# exit</pre>	Exits LISP site configuration mode and returns to LISP configuration mode.
Step 7	Repeat steps 3 through 6 for each LISP site to be configured.	In this example, steps 3 through 6 would be repeated for the site SOC and FIN as illustrated in the complete configuration example at the end of this task.
Step 8	<p>ipv4 map-resolver</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 map-resolver</pre>	Enables LISP map resolver functionality for EIDs in the IPv4 address family.
Step 9	<p>ipv4 map-server</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 map-server</pre>	Enables LISP map server functionality for EIDs in the IPv4 address family.

	Command or Action	Purpose
Step 10	<p>eid-table vrfvrf-name instance-id instance-id</p> <p>Example:</p> <pre>Router(config-router-lisp)# eid-table vrf TRANS instance-id 1</pre>	<p>Configures an association between a VRF table and a LISP instance ID, and enters eid-table configuration submode.</p> <ul style="list-style-type: none"> In this example, the VRF table TRANS and instance-id 1 are associated together.
Step 11	<p>database-mapping EID-prefix/prefix-length locator priority priority weight weight</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# database-mapping 10.1.1.0/24 172.16.1.2 priority 1 weight 100</pre>	<p>Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site.</p> <ul style="list-style-type: none"> In this example, the EID prefix 10.1.1.0/24 within instance-id 1 at this site is associated with the local IPv4 RLOC 172.16.1.2, as well as with the neighbor xTR RLOC 172.6.1.6.
Step 12	<p>Repeat Step 11 until all EID-to-RLOC mappings within this eid-table vrf and instance ID for the LISP site are configured.</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# database-mapping 10.1.1.0/24 172.16.1.6 priority 1 weight 100</pre>	<p>Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site.</p>
Step 13	<p>ipv4 etr map-server map-server-address key key-type authentication-key</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# ipv4 etr map-server 172.16.1.2 key 0 TRANS-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key for which this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> In this example, the map server and authentication-key are specified here, within the eid-table subcommand mode, so that the authentication key is associated only with this instance ID, within this VPN. <p>Note The map server must be configured with EID prefixes and instance-ids matching the one(s) configured on this ETR, as well as an identical authentication key.</p> <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map server is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 14	<p>Repeat Step 13 to configure another locator address for the same LISP map server</p> <p>Example:</p>	<p>Configures a locator address for the LISP map server and an authentication key for which this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p>

	Command or Action	Purpose
	<pre>Router(config-router-lisp-eid-table)# ipv4 etr map-server 172.16.1.6 key 0 TRANS-key</pre>	<ul style="list-style-type: none"> In this example, a redundant map server is configured. (Because the MS is co-located with the xTRs in this case, this command indicates that this xTR is pointing to itself for registration (and its neighbor xTR/MS/MR at the same site).
Step 15	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# exit</pre>	Exits eid-table configuration submode and returns to LISP configuration mode.
Step 16	<p>ipv4 itr map-resolver map-resolver-address</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 172.16.1.2</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> In this example, the map resolver is specified within router lisp configuration mode and inherited into all eid-table instances since nothing is related to any single instance ID. In addition, redundant map resolvers are configured. (Because the MR is co-located with the xTRs in this case, this command indicates that this xTR is pointing to itself for mapping resolution (and its neighbor xTR/MS/MR at the same site). The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable using its IPv4 locator address. (See the <i>LISP Command Reference Guide</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 17	<p>Repeat Step 16 to configure another locator address for the LISP map resolver</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 172.16.1.6</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> In this example, a redundant map resolver is configured. (Because the MR is co-located with the xTRs in this case, this command indicates that this xTR is pointing to itself for mapping resolution (and its neighbor xTR/MS/MR at the same site). The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable using its IPv4 locator address.

	Command or Action	Purpose
		<p>(See the <i>LISP Command Reference Guide</i> for more details.)</p> <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 18	<p>ipv4 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr</pre>	Enables LISP ITR functionality for the IPv4 address family.
Step 19	<p>ipv4 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr</pre>	Enables LISP ETR functionality for the IPv4 address family.
Step 20	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 21	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 172.16.1.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> LISP-encapsulated to a LISP site when traffic is LISP-to-LISP natively forwarded when traffic is LISP-to-non-LISP Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP EID and the destination matches one of the following entries: <ul style="list-style-type: none"> a current map-cache entry a default route with a legitimate next-hop no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
Step 22	<p>exit</p> <p>Example:</p>	Exits global configuration mode.

	Command or Action	Purpose
	Router(config)# exit	

Example:

The examples below show the complete configuration for the HQ-RTR-1 and HQ-RTR-2 (xTR/MS/MR located at the HQ Site), and Site2-xTR LISP devices illustrated in the figure above and in this task. Note that both HQ-RTR-1 and HQ-RTR-2 are provided in order to illustrate the proper method for configuring a LISP multihomed site.

Example configuration for HQ-RTR-1 with an xTR, a map server and a map resolver:

```

hostname HQ-RTR-1
!
vrf definition TRANS
  address-family ipv4
  exit
!
vrf definition SOC
  address-family ipv4
  exit
!
vrf definition FIN
  address-family ipv4
  exit
!
interface Loopback0
  description Management Loopback (in default space)
  ip address 172.31.1.11 255.255.255.255
!
interface GigabitEthernet0/0/0
  description WAN Link to IPv4 Core
  ip address 172.16.1.2 255.255.255.252
  negotiation auto
!
interface GigabitEthernet0/0/1
  vrf forwarding TRANS
  ip address 10.1.1.1 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/0/2
  vrf forwarding SOC
  ip address 10.2.1.1 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/0/3
  vrf forwarding FIN
  ip address 10.3.1.1 255.255.255.0
  negotiation auto
!
router lisp
  eid-table default instance-id 0
    database-mapping 172.31.1.11/32 172.16.1.2 priority 1 weight 50
    database-mapping 172.31.1.11/32 172.16.1.6 priority 1 weight 50
    ipv4 etr map-server 172.16.1.2 key DEFAULT-key
    ipv4 etr map-server 172.16.1.6 key DEFAULT-key
  exit
!
  eid-table vrf TRANS instance-id 1

```

```

database-mapping 10.1.1.0/24 172.16.1.2 priority 1 weight 50
database-mapping 10.1.1.0/24 172.16.1.6 priority 1 weight 50
ipv4 etr map-server 172.16.1.2 key TRANS-key
ipv4 etr map-server 172.16.1.6 key TRANS-key
exit
!
eid-table vrf SOC instance-id 2
database-mapping 10.2.1.0/24 172.16.1.2 priority 1 weight 50
database-mapping 10.2.1.0/24 172.16.1.6 priority 1 weight 50
ipv4 etr map-server 172.16.1.2 key SOC-key
ipv4 etr map-server 172.16.1.6 key SOC-key
exit
!
eid-table vrf FIN instance-id 3
database-mapping 10.3.1.0/24 172.16.1.2 priority 1 weight 50
database-mapping 10.3.1.0/24 172.16.1.6 priority 1 weight 50
ipv4 etr map-server 172.16.1.2 key FIN-key
ipv4 etr map-server 172.16.1.6 key FIN-key
exit
!
site DEFAULT
authentication-key DEFAULT-key
eid-prefix 172.31.1.0/24 accept-more-specifics
exit
!
site TRANS
authentication-key TRANS-key
eid-prefix instance-id 1 10.1.0.0/16 accept-more-specifics
exit
!
site SOC
authentication-key SOC-key
eid-prefix instance-id 2 10.2.0.0/16 accept-more-specifics
exit
!
site FIN
authentication-key FIN-key
eid-prefix instance-id 3 10.3.0.0/16 accept-more-specifics
exit
!
ipv4 map-server
ipv4 map-resolver
ipv4 itr map-resolver 172.16.1.2
ipv4 itr map-resolver 172.16.1.6
ipv4 itr
ipv4 etr
exit
!
ip route 0.0.0.0 0.0.0.0 172.16.1.1

```

Example configuration for HQ-RTR-2 with an xTR, a map server and a map resolver:

```

hostname HQ-RTR-2
!
vrf definition TRANS
address-family ipv4
exit
!
vrf definition SOC
address-family ipv4
exit
!
vrf definition FIN
address-family ipv4

```

```
exit
!
interface Loopback0
  description Management Loopback (in default space)
  ip address 172.31.1.12 255.255.255.255
!
interface GigabitEthernet0/0/0
  description WAN Link to IPv4 Core
  ip address 172.16.1.6 255.255.255.252
  negotiation auto
!
interface GigabitEthernet0/0/1
  vrf forwarding TRANS
  ip address 10.1.1.2 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/0/2
  vrf forwarding SOC
  ip address 10.2.1.2 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/0/3
  vrf forwarding FIN
  ip address 10.3.1.2 255.255.255.0
  negotiation auto
!
router lisp
  eid-table default instance-id 0
    database-mapping 172.31.1.12/32 172.16.1.2 priority 1 weight 50
    database-mapping 172.31.1.12/32 172.16.1.6 priority 1 weight 50
    ipv4 etr map-server 172.16.1.2 key DEFAULT-key
    ipv4 etr map-server 172.16.1.6 key DEFAULT-key
  exit
!
  eid-table vrf TRANS instance-id 1
    database-mapping 10.1.1.0/24 172.16.1.2 priority 1 weight 50
    database-mapping 10.1.1.0/24 172.16.1.6 priority 1 weight 50
    ipv4 etr map-server 172.16.1.2 key TRANS-key
    ipv4 etr map-server 172.16.1.6 key TRANS-key
  exit
!
  eid-table vrf SOC instance-id 2
    database-mapping 10.2.1.0/24 172.16.1.2 priority 1 weight 50
    database-mapping 10.2.1.0/24 172.16.1.6 priority 1 weight 50
    ipv4 etr map-server 172.16.1.2 key SOC-key
    ipv4 etr map-server 172.16.1.6 key SOC-key
  exit
!
  eid-table vrf FIN instance-id 3
    database-mapping 10.3.1.0/24 172.16.1.2 priority 1 weight 50
    database-mapping 10.3.1.0/24 172.16.1.6 priority 1 weight 50
    ipv4 etr map-server 172.16.1.2 key FIN-key
    ipv4 etr map-server 172.16.1.6 key FIN-key
  exit
!
  site DEFAULT
    authentication-key DEFAULT-key
    eid-prefix 172.31.1.0/24 accept-more-specifics
  exit
!
  site TRANS
    authentication-key TRANS-key
    eid-prefix instance-id 1 10.1.0.0/16 accept-more-specifics
  exit
```

Configure a Remote Site for Large-Scale LISP Shared Model Virtualization

```

!
site SOC
 authentication-key SOC-key
 eid-prefix instance-id 2 10.2.0.0/16 accept-more-specifics
 exit
!
site FIN
 authentication-key FIN-key
 eid-prefix instance-id 3 10.3.0.0/16 accept-more-specifics
 exit
!
ipv4 map-server
ipv4 map-resolver
ipv4 itr map-resolver 172.16.1.2
ipv4 itr map-resolver 172.16.1.6
ipv4 itr
ipv4 etr
exit
!
ip route 0.0.0.0 0.0.0.0 172.16.1.5

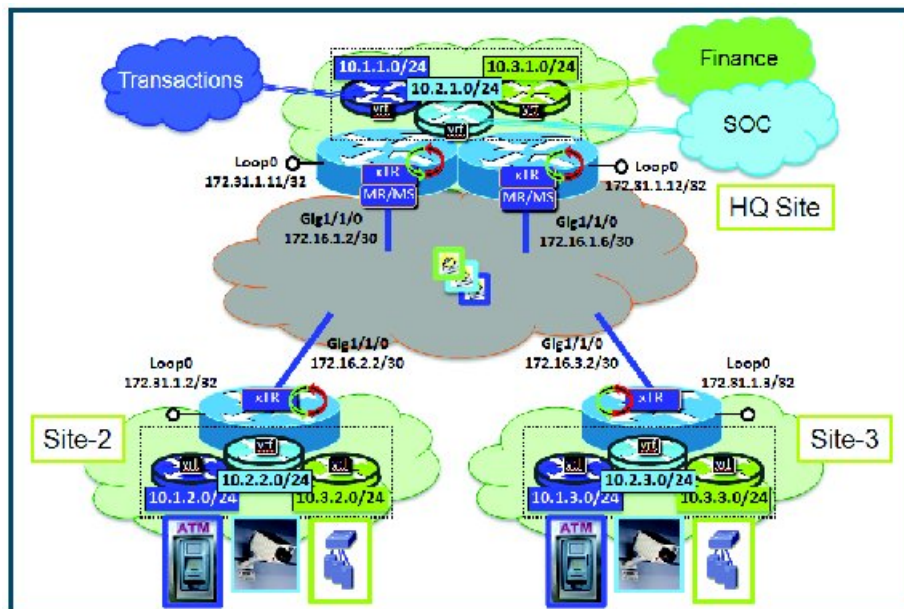
```

Configure a Remote Site for Large-Scale LISP Shared Model Virtualization

Perform this task to enable and configure LISP ITR/ETR (xTR) functionality at a remote site to implement LISP shared model virtualization as part of a large-scale, multiple-site LISP topology.

The configuration demonstrated in this task is part of a more complex, larger scale LISP virtualization solution. In this task, the configuration applies to one of the remote sites shown in the figure below. In this task, the remote site routers only act as xTRs, and use the MS/MRs at the HQ site for LISP control plane support.

Figure 30: Large Scale LISP Site with Virtualized IPv4 EIDs and a Shared IPv4 Core



The components illustrated in the topology shown in the figure above are described below:

- **LISP remote sites:**
 - Each remote site CPE router functions as a LISP ITR and ETR (xTR).

- Each LISP xTRs has the same three VRFs as the HQ Site: TRANS, SOC, and FIN. Each VRF contains only IPv4 EID-prefixes.
- Each remote site LISP xTR has a single RLOC connection to a shared IPv4 core network.

Before you begin

The configuration below assumes that the referenced VRFs were created using the **vrf definition** command and that the Configure a Large-Scale LISP Shared Model Virtualization task has been performed at one or more central (headquarters) sites.

SUMMARY STEPS

1. **configure terminal**
2. **router lisp**
3. **eid-table vrfvrf-name instance-id instance-id**
4. **database-mapping EID-prefix/prefix-length locator priority priority weight weight**
5. **ipv4 etr map-server map-server-address key key-type authentication-key**
6. Repeat Step 13 to configure another locator address for the same LISP map server
7. **exit**
8. **ipv4 itr map-resolver map-resolver-address**
9. Repeat Step 16 to configure another locator address for the LISP map resolver
10. **ipv4 itr**
11. **ipv4 etr**
12. **exit**
13. **ip route ipv4-prefix next-hop**
14. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 2	router lisp Example: <pre>Router(config)# router lisp</pre>	Enters LISP configuration mode (software only).
Step 3	eid-table vrfvrf-name instance-id instance-id Example: <pre>Router(config-router-lisp)# eid-table vrf TRANS instance-id 1</pre>	Configures an association between a VRF table and a LISP instance ID, and enters eid-table configuration submode. <ul style="list-style-type: none"> • In this example, the VRF table TRANS and instance-id 1 are associated together.

	Command or Action	Purpose
Step 4	<p>database-mapping <i>EID-prefix/prefix-length locator priority priority weight weight</i></p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# database-mapping 10.1.2.0/24 172.16.2.2 priority 1 weight 100</pre>	<p>Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site.</p> <ul style="list-style-type: none"> In this example, the EID prefix 10.1.2.0/24 within instance-id 1 at this site is associated with the local IPv4 RLOC 172.16.2.2. <p>Note Repeat this step until all EID-to-RLOC mappings within this eid-table vrf and instance ID for the LISP site are configured.</p>
Step 5	<p>ipv4 etr map-server <i>map-server-address key key-type authentication-key</i></p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# ipv4 etr map-server 172.16.1.2 key 0 TRANS-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key for which this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> In this example, the map server and authentication-key are specified here, within the eid-table subcommand mode, so that the authentication key is associated only with this instance ID, within this VPN. <p>Note The map server must be configured with EID prefixes and instance-ids matching the one(s) configured on this ETR, as well as an identical authentication key.</p> <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map server is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 6	<p>Repeat Step 13 to configure another locator address for the same LISP map server</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# ipv4 etr map-server 172.16.1.6 key 0 TRANS-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key for which this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> In this example, a redundant map server is configured. (Because the MS is co-located with the xTRs in this case, this command indicates that this xTR is pointing to itself for registration (and its neighbor xTR/MS/MR at the same site).
Step 7	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# exit</pre>	<p>Exits eid-table configuration submode and returns to LISP configuration mode.</p>

	Command or Action	Purpose
Step 8	<p>ipv4 itr map-resolver <i>map-resolver-address</i></p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 172.16.1.2</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> In this example, the map resolver is specified within router lisp configuration mode and inherited into all eid-table instances since nothing is related to any single instance ID. In addition, redundant map resolvers are configured. (Because the MR is co-located with the xTRs in this case, this command indicates that this xTR is pointing to itself for mapping resolution (and its neighbor xTR/MS/MR at the same site). The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable using its IPv4 locator address. (See the <i>LISP Command Reference Guide</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 9	<p>Repeat Step 16 to configure another locator address for the LISP map resolver</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 172.16.1.6</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> In this example, a redundant map resolver is configured. (Because the MR is co-located with the xTRs in this case, this command indicates that this xTR is pointing to itself for mapping resolution (and its neighbor xTR/MS/MR at the same site). The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable using its IPv4 locator address. (See the <i>LISP Command Reference Guide</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 10	<p>ipv4 itr</p> <p>Example:</p>	<p>Enables LISP ITR functionality for the IPv4 address family.</p>

	Command or Action	Purpose
	<pre>Router(config-router-lisp)# ipv4 itr</pre>	
Step 11	<p>ipv4 etr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr</pre>	Enables LISP ETR functionality for the IPv4 address family.
Step 12	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 13	<p>ip route <i>ipv4-prefix next-hop</i></p> <p>Example:</p> <pre>Router(config)# ip route 0.0.0.0 0.0.0.0 172.16.2.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> • All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> • LISP-encapsulated to a LISP site when traffic is LISP-to-LISP • natively forwarded when traffic is LISP-to-non-LISP • Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP EID and the destination matches one of the following entries: <ul style="list-style-type: none"> • a current map-cache entry • a default route with a legitimate next-hop • no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
Step 14	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	Exits global configuration mode.

Example:

The example below show the complete configuration for the remote site device illustrated in the figure above and in this task. Note that only one remote site configuration is shown here.

Example configuration for Site 2 with an xTR, and using the map server and a map resolver from the HQ site:


```
hostname Site2-xTR
!
vrf definition TRANS
address-family ipv4
  exit
!
vrf definition SOC
address-family ipv4
  exit
!
vrf definition FIN
address-family ipv4
  exit
!
interface Loopback0
  description Management Loopback (in default space)
  ip address 172.31.1.2 255.255.255.255
!
interface GigabitEthernet0/0/0
  description WAN Link to IPv4 Core
  ip address 172.16.2.2 255.255.255.252
  negotiation auto
!
interface GigabitEthernet0/0/1
  vrf forwarding TRANS
  ip address 10.1.2.1 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/0/2
  vrf forwarding SOC
  ip address 10.2.2.1 255.255.255.0
  negotiation auto
!
interface GigabitEthernet0/0/3
  vrf forwarding FIN
  ip address 10.3.2.1 255.255.255.0
  negotiation auto
!
router lisp
  eid-table default instance-id 0
    database-mapping 172.31.1.2/32 172.16.2.2 priority 1 weight 100
    ipv4 etr map-server 172.16.1.2 key DEFAULT-key
    ipv4 etr map-server 172.16.1.6 key DEFAULT-key
    exit
  !
  eid-table vrf TRANS instance-id 1
    database-mapping 10.1.2.0/24 172.16.2.2 priority 1 weight 100
    ipv4 etr map-server 172.16.1.2 key TRANS-key
    ipv4 etr map-server 172.16.1.6 key TRANS-key
    exit
  !
  eid-table vrf SOC instance-id 2
    database-mapping 10.2.2.0/24 172.16.2.2 priority 1 weight 100
    ipv4 etr map-server 172.16.1.2 key SOC-key
    ipv4 etr map-server 172.16.1.6 key SOC-key
    exit
  !
  eid-table vrf FIN instance-id 3
    database-mapping 10.3.2.0/24 172.16.2.2 priority 1 weight 100
    ipv4 etr map-server 172.16.1.2 key FIN-key
    ipv4 etr map-server 172.16.1.6 key FIN-key
    exit
  !
```

```

ipv4 itr map-resolver 172.16.1.2
ipv4 itr map-resolver 172.16.1.6
ipv4 itr
ipv4 etr
exit
!
ip route 0.0.0.0 0.0.0.0 172.16.2.1

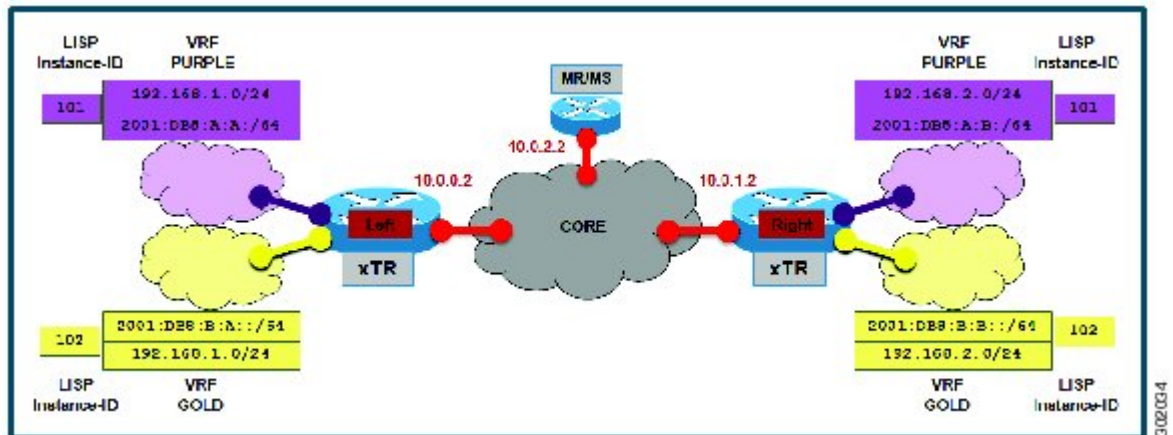
```

Verifying and Troubleshooting LISP Virtualization

After configuring LISP, verifying and troubleshooting LISP configuration and operations may be performed by following the optional steps described below. Note that certain verification and troubleshooting steps may only apply to certain types of LISP devices.

In this task, the topology is shown in the figure below and the configuration is from the “Configure Simple LISP Shared Model Virtualization” task, but the commands are applicable to both LISP shared and parallel model virtualization.

Figure 31: Simple LISP Site with Virtualized IPv4 and IPv6 EIDs and a Shared IPv4 Core



Note

The following examples do not show every available command and every available output display. Refer to the *Cisco IOS LISP Command Reference* for detailed explanations of each command.

SUMMARY STEPS

1. **enable**
2. **show running-config | section router lisp**
3. **show [ip | ipv6] lisp**
4. **show [ip | ipv6] lisp map-cache**
5. **show [ip | ipv6] lisp database [eid-table vrf vrf-name]**
6. **show lisp site [name site-name]**
7. **lig [{self {ipv4 | ipv6}}] | {hostname | destination-EID}**
8. **ping {hostname | destination-EID}**
9. **clear [ip | ipv6] lisp map-cache**

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 show running-config | section router lisp

The **show running-config | section router lisp** command is useful for quickly verifying the LISP configuration on the device. This command applies to any LISP device. The following is sample output from the **show running-config | section router lisp** command when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and a shared IPv4 core:

Example:

```
Router# show running-config | section router lisp

router lisp
  eid-table vrf PURPLE instance-id 101
    database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
    database-mapping 2001:DB8:A:A::/64 10.0.0.2 priority 1 weight 1
  eid-table vrf GOLD instance-id 102
    database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
    database-mapping 2001:DB8:B:A::/64 10.0.0.2 priority 1 weight 1
  exit
  !
  ipv4 itr map-resolver 10.0.2.2
  ipv4 itr
  ipv4 etr map-server 10.0.2.2 key Left-key
  ipv4 etr
  ipv6 itr map-resolver 10.0.2.2
  ipv6 itr
  ipv6 etr map-server 10.0.2.2 key Left-key
  ipv6 etr
  exit
```

Step 3 show [ip | ipv6] lisp

The **show ip lisp** and **show ipv6 lisp** commands are useful for quickly verifying the operational status of LISP as configured on the device, as applicable to the IPv4 and IPv6 address families respectively. This command applies to any LISP device.

Example:

The first example shows a summary of LISP operational status and IPv6 address family information by EID table:

```
Router# show ipv6 lisp eid-table summary

Instance count: 2
Key: DB - Local EID Database entry count (@ - RLOC check pending
      * - RLOC consistency problem),
     DB no route - Local EID DB entries with no matching RIB route,
     Cache - Remote EID mapping cache size, IID - Instance ID,
     Role - Configured Role

Interface      DB  DB no  Cache Incom Cache
```

EID VRF name	(.IID)	size	route	size	plete	Idle	Role
PURPLE	LISPO.101	1	0	1	0.0%	0.0%	ITR-ETR
GOLD	LISPO.102	1	0	1	0.0%	0.0%	ITR-ETR

Example:

The second example shows LISP operational status and IPv6 address family information for the VRF named PURPLE:

```
Router# show ipv6 lisp eid-table vrf PURPLE

Instance ID:                101
Router-lisp ID:             0
Locator table:              default
EID table:                  PURPLE
Ingress Tunnel Router (ITR): enabled
Egress Tunnel Router (ETR): enabled
Proxy-ITR Router (PITR):   disabled
Proxy-ETR Router (PETR):   disabled
Map Server (MS):           disabled
Map Resolver (MR):         disabled
Map-Request source:        2001:DB8:A:A::1
ITR Map-Resolver(s):       10.0.2.2
ETR Map-Server(s):         10.0.2.2 (00:00:24)
ITR use proxy ETR RLOC(s): none
```

Example:

The third example shows LISP operational status and IPv6 address family information for the instance ID of 101:

```
Router# show ipv6 lisp instance-id 101

Instance ID:                101
Ingress Tunnel Router (ITR): enabled
Egress Tunnel Router (ETR): enabled
Proxy-ITR Router (PITR):   disabled
Proxy-ETR Router (PETR):   disabled
Map Server (MS):           disabled
Map Resolver (MR):         disabled
Map-Request source:        2001:DB8:A:A::1
ITR Map-Resolver(s):       10.0.2.2
ETR Map-Server(s):         10.0.2.2 (00:00:11)
ITR Solicit Map Request (SMR): accept and process
  Max SMRs per map-cache entry: 8 more specifics
  Multiple SMR suppression time: 60 secs
ETR accept mapping data:   disabled, verify disabled
ETR map-cache TTL:         1d00h
```

Step 4 `show [ip | ipv6] lisp map-cache`

The `show ip lisp map-cache` and `show ipv6 lisp map-cache` commands are useful for quickly verifying the operational status of the map cache on a device configured as an ITR or PITR, as applicable to the IPv4 and IPv6 address families respectively.

Example:

The following example shows IPv6 mapping cache information based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and a shared IPv4 core. This example output assumes that a map-cache entry has been received for another site with the IPv6 EID prefix 2001:db8:b:b::/64.

```
Router# show ip lisp map-cache eid-table vrf GOLD
```

```
LISP IPv6 Mapping Cache for EID-table vrf GOLD (IID 102), 2 entries
::/0, uptime: 01:09:52, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
2001:DB8:B:B::/64, uptime: 00:00:10, expires: 23:59:42, via map-reply, complete
  Locator    Uptime    State    Pri/Wgt
  10.0.1.2   00:00:10  up      1/1
```

Step 5 `show [ip | ipv6] lisp database [eid-table vrf vrf-name]`

The `show ip lisp database` and `show ipv6 lisp database` commands are useful for quickly verifying the operational status of the database mapping on a device configured as an ETR, as applicable to the IPv4 and IPv6 address families respectively.

Example:

The following example shows IPv6 mapping database information for the VRF named GOLD.

```
Router# show ipv6 lisp database eid-table vrf GOLD

LISP ETR IPv6 Mapping Database for EID-table vrf GOLD (IID 102), LSBs: 0x1, 1 entries

EID-prefix: 2001:DB8:B:A::/64
  10.0.0.2, priority: 1, weight: 1, state: site-self, reachable
```

Step 6 `show lisp site [name site-name]`

The `show lisp site` command is useful for quickly verifying the operational status of LISP sites, as configured on a map server. This command only applies to a device configured as a map server. The following example output is based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and shows the information for the instance ID of 101.

Example:

```
Router# show lisp site instance-id 101

LISP Site Registration Information

Site Name      Last      Up    Who Last      Inst  EID Prefix
              Register  yes   Registered   ID
Left           00:00:36 yes   10.0.0.2     101   192.168.1.0/24
              00:00:43 yes   10.0.0.2     101   2001:DB8:A:A::/64
Right          00:00:31 yes   10.0.1.2     101   192.168.2.0/24
              00:00:02 yes   10.0.1.2     101   2001:DB8:A:B::/64
```

Example:

This second example shows LISP site information for the IPv6 EID prefix of 2001:db8:a:a:/64 and instance ID of 101.

```
Router# show lisp site 2001:db8:a:a:/64 instance-id 101

LISP Site Registration Information

Site name: Left
Allowed configured locators: any
Requested EID-prefix:
  EID-prefix: 2001:DB8:A:A::/64 instance-id 101
  First registered:    02:41:55
  Routing table tag:  0
  Origin:              Configuration
Registration errors:
  Authentication failures: 4
  Allowed locators mismatch: 0
```

```

ETR 10.0.0.2, last registered 00:00:22, no proxy-reply, no map-notify
      TTL 1d00h
Locator  Local  State      Pri/Wgt
10.0.0.2  yes    up          1/1

```

Step 7 **lig** {[self {ipv4 | ipv6}] | {hostname | destination-EID}}

The LISP Internet Groper (**lig**) command is useful for testing the LISP control plane. The **lig** command can be used to query for the indicated destination hostname or EID, or the routers local EID-prefix. This command provides a simple means of testing whether a destination EID exists in the LISP mapping database system, or your site is registered with the mapping database system. This command is applicable for both the IPv4 and IPv6 address families and applies to any LISP device that maintains a map cache (for example, if configured as an ITR or PITR). The following example output is based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and shows the information for the instance ID of 101 and the IPv4 EID prefix of 192.168.2.1.

Example:

```
Router# lig instance-id 101 192.168.2.1
```

```

Mapping information for EID 192.168.2.1 from 10.0.1.2 with RTT 12 msec
192.168.2.0/24, uptime: 00:00:00, expires: 23:59:52, via map-reply, complete
Locator  Uptime   State      Pri/Wgt
10.0.1.2 00:00:00 up          1/1

```

Example:

This second example output shows information about the VRF named PURPLE:

```
Router# lig eid-table vrf PURPLE self
```

```

Mapping information for EID 192.168.1.0 from 10.0.0.1 with RTT 20 msec
192.168.1.0/24, uptime: 00:00:00, expires: 23:59:52, via map-reply, self
Locator  Uptime   State      Pri/Wgt
10.0.0.1 00:00:00 up, self    1/1

```

Step 8 **ping** {hostname | destination-EID}

The **ping** command is useful for testing basic network connectivity and reachability and/or liveness of a destination EID or RLOC address. When using **ping** it is important to be aware that because LISP uses an encapsulation, you should always specify a source address; never allow the **ping** application to assign its own default source address. This is because there are four possible ways to use **ping**, and without explicitly indicating the source address, the wrong one may be used by the application leading to erroneous results that complicate operational verification or troubleshooting. The four possible uses of **ping** include:

- RLOC-to-RLOC—Sends “echo” packets out natively (no LISP encaps) and receive the “echo-reply” back natively. This can be used to test the underlying network connectivity between locators of various devices, such as xTR to Map-Server or Map-Resolver.
- EID-to-EID—Sends “echo” packets out LISP-encaped and receive the “echo-reply” back LISP-encaped. This can be used to test the LISP data plane (encapsulation) between LISP sites.
- EID-to-RLOC—Sends “echo” packets out natively (no LISP encaps) and receive the "echo-reply" back LISP-encaped through a PITR mechanism. This can be used to test the PITR infrastructure.
- RLOC-to-EID - Sends “echo” packets out LISP-encaped and receive the “echo-reply” back natively. This can be used to test PETR capabilities.

The **ping** command is applicable to the IPv4 and IPv6 address families respectively, and can be used on any LISP device in some manner. (The ability to do LISP encapsulation, for example, requires the device to be configured as an ITR or PITR.)

The following example output from the **ping** command is based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes. (Note that ping is not a LISP command and does not know about an EID table or an instance ID. When virtualization is included, output limiters can only be specified by VRF.)

Example:

```
Router# ping vrf PURPLE 2001:DB8:a:b::1 source 2001:DB8:a:a::1 rep 100

Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 2001:DB8:A:B::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:A:A::1%PURPLE
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (100/100), round-trip min/avg/max = 0/0/1 ms
```

Example:

```
Router# ping vrf GOLD

Protocol [ip]: ipv6
Target IPv6 address: 2001:db8:b:b::1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands? [no]: y
Source address or interface: 2001:db8:b:a::1
.
.
.
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:B:B::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:B:A::1%GOLD
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
```

Step 9 clear [ip | ipv6] lisp map-cache

The **clear ip lisp map-cache** and **clear ipv6 lisp map-cache** commands remove all IPv4 or IPv6 dynamic LISP map-cache entries stored by the router. This can be useful trying to quickly verify the operational status of the LISP control plane. This command applies to a LISP device that maintains a map cache (for example, if configured as an ITR or PITR).

Example:

The following example displays IPv4 mapping cache information for instance ID 101, shows the command used to clear the mapping cache for instance ID 101, and displays the show information after clearing the cache.

```
Router# show ip lisp map-cache instance-id 101

LISP IPv4 Mapping Cache for EID-table vrf PURPLE (IID 101), 2 entries

0.0.0.0/0, uptime: 00:25:17, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
192.168.2.0/24, uptime: 00:20:13, expires: 23:39:39, via map-reply, complete
  Locator   Uptime    State     Pri/Wgt
  10.0.1.2  00:20:13  up        1/1

Router# clear ip lisp map-cache instance-id 101
```

```
Router# show ip lisp map-cache instance-id 101

LISP IPv4 Mapping Cache, 1 entries

0.0.0.0/0, uptime: 00:00:02, expires: never, via static send map-request
Negative cache entry, action: send-map-request
```

Configuration Examples for LISP Shared Model Virtualization

Complete configuration examples are available within each task under the “How to Configure LISP Shared Model Virtualization” section.

Additional References

Related Documents

Document Title	Location
Cisco IOS IP Routing: LISP Command Reference	http://www.cisco.com/en/US/docs/ios-xml/ios/iproute_lisp/command/ip-lisp-cr-book.html
Enterprise IPv6 Transitions Strategy Using the Locator/ID Separation Protocol	Cisco LISP Software Image Download Page
Cisco IOS LISP0 Virtual Interface, Application Note, Version 1.0	Cisco LISP Software Image Download Page
Cross-Platform Release Notes for Cisco IOS Release 15.2M&T	http://www.cisco.com/en/US/docs/ios/15_2m_and_t/release/notes/15_2m_and_t.html

Standards

Standard	Title
IANA Address Family Numbers	http://www.iana.org/assignments/address-family-numbers/address-family-numbers.xml

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
draft-ietf-lisp-22	Locator/ID Separation Protocol (LISP) http://tools.ietf.org/html/draft-ietf-lisp-22
draft-ietf-lisp-ms-16	LISP Map Server http://tools.ietf.org/html/draft-ietf-lisp-ms-16
draft-ietf-lisp-alt-10	LISP Alternative Topology (LISP+ALT) http://tools.ietf.org/html/draft-ietf-lisp-alt-10
draft-ietf-lisp-LCAF-06	LISP Canonical Address Format (LCAF) http://tools.ietf.org/wg/lisp/
draft-ietf-lisp-interworking-06	Interworking LISP with IPv4 and IPv6 http://tools.ietf.org/html/draft-ietf-lisp-interworking-06
draft-ietf-lisp-lig-06	LISP Internet Groper (LIG) http://tools.ietf.org/html/draft-ietf-lisp-lig-06
draft-ietf-lisp-mib-03	LISP MIB http://tools.ietf.org/wg/lisp/draft-ietf-lisp-mib/

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP Shared Model Virtualization

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 4: Feature Information for LISP Shared Model Virtualization

Feature Name	Releases	Feature Information
LISP Shared Model Virtualization	15.2(2)T 15.1(1)SY1	LISP Shared Model Virtualization feature uses Endpoint Identifier (EID) spaces that are created by binding VRFs associated with an EID space to Instance IDs. A common, “shared” locator space is used by all virtualized EIDs.



CHAPTER 6

LISP Parallel Model Virtualization

- [Finding Feature Information](#), on page 145
- [Information About LISP Parallel Model Virtualization](#), on page 145
- [How to Configure LISP Parallel Model Virtualization](#), on page 150
- [Configuration Examples for LISP Parallel Model Virtualization](#), on page 168
- [Additional References](#), on page 168
- [Feature Information for LISP Parallel Model Virtualization](#), on page 169

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

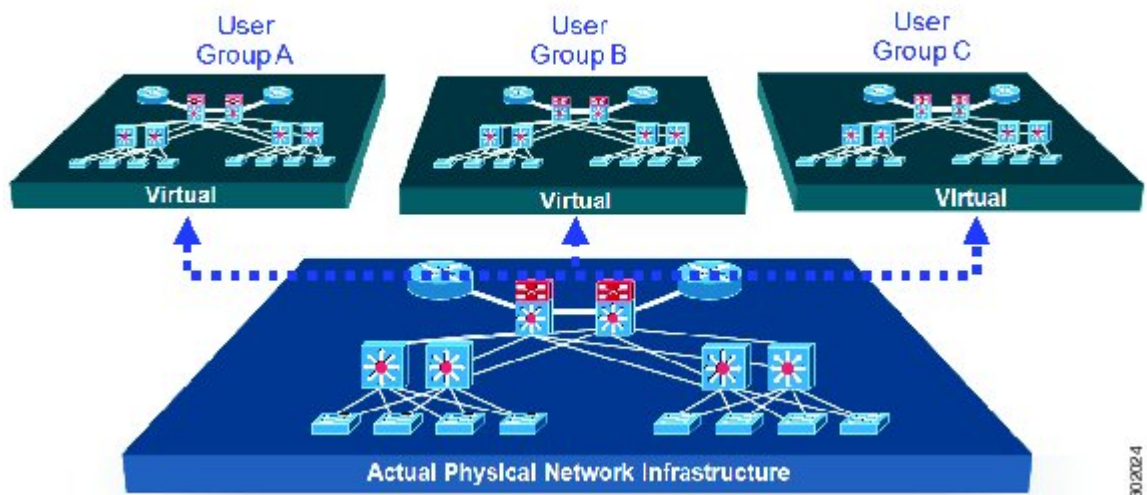
Information About LISP Parallel Model Virtualization

Overview of LISP Virtualization

Deploying physical network infrastructure requires both capital investments for hardware, as well as manpower investments for installation and operational management support. When distinct user groups within an organization desire to control their own networks, it rarely makes economic sense for these user groups to deploy and manage separate physical networks. Physical plants are rarely utilized to their fullest, resulting in stranded capacity (bandwidth, processor, memory, etc.). In addition, the power, rack space, and cooling needs to physical plants do not satisfy modern “green” requirements. Network virtualization offers the opportunity to satisfy organizational needs, while efficiently utilizing physical assets.

The purpose of network virtualization, as shown in the figure below, is to create multiple, logically separated topologies across one common physical infrastructure.

Figure 32: LISP Deployment Environment



When considering the deployment of a virtualized network environment, take into account both the device and the path level.

Device Level Virtualization

Virtualization at the device level entails the use of the virtual routing and forwarding (VRF) to create multiple instances of Layer 3 routing tables, as illustrated in the figure below. VRFs provide segmentation across IP addresses, allowing for overlapped address space and traffic separation. Separate routing, QoS, security, and management policies can be applied to each VRF instance. An IGP or EGP routing process is typically enabled within a VRF, just as it would be in the global (default) routing table. As described in detail below, LISP binds VRFs to instance IDs for similar purposes.

Figure 33: Device Level Virtualization

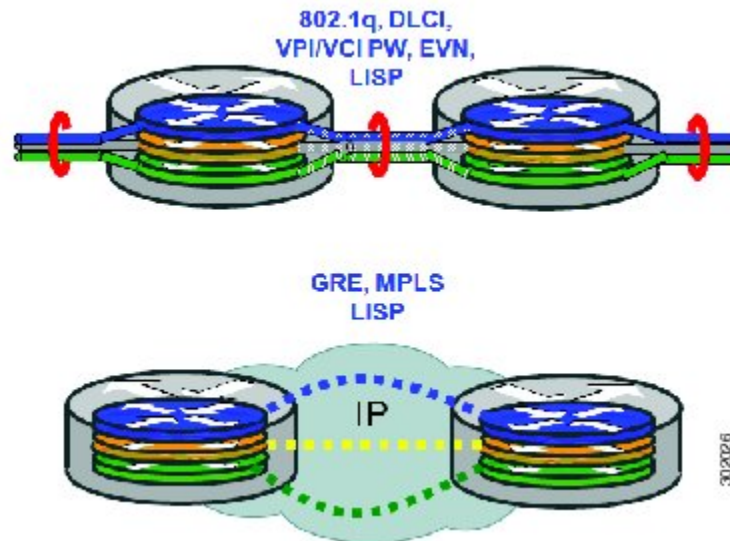


Path Level Virtualization

VRF table separation is maintained across network paths using any number of traditional mechanisms, as illustrated in the figure below. Single-hop path segmentation (hop-by-hop) is typically accomplished by techniques such as 802.1q VLANs, VPI/VCI PW, or EVN. LISP can also be used. Traditional multi-hop

mechanisms include MPLS and GRE tunnels. As described in detail below, LISP binds VRFs to instance IDs (IIDs), and then these IIDs are included in the LISP header to provide data plane (traffic flow) separation for single or multihop needs.

Figure 34: Path Level Virtualization



LISP Virtualization at the Device Level

Recalling that LISP implements Locator ID separation and, in so doing, creates two namespaces (EIDs and RLOCs), it is easy to see that LISP virtualization can consider both EID and RLOC namespaces for virtualization. That is, either or both can be virtualized.

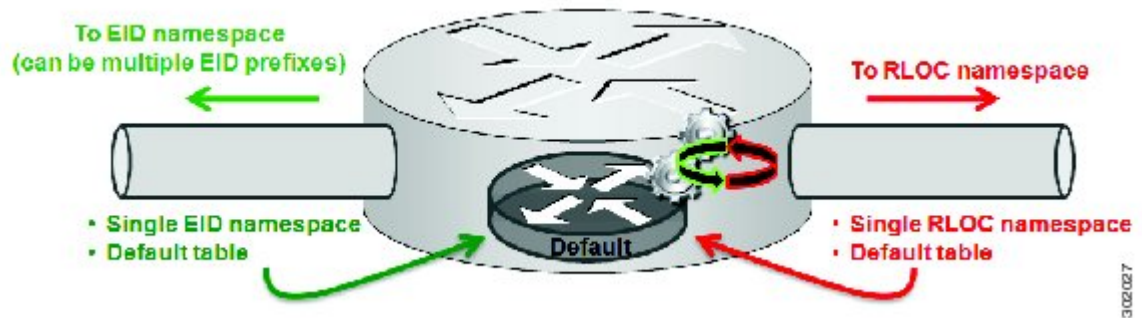
- EID virtualization—Enabled by binding a LISP instance ID to an EID VRF. Instance IDs are numerical tags defined in the LISP canonical address format (LCAF) draft, and are used to maintain address space segmentation in both the control plane and data plane.
- RLOC virtualization—Tying locator addresses and associated mapping services to the specific VRF within which they are reachable enables RLOC virtualization.

Because LISP considers virtualization of both EID and RLOC namespaces, two models of operation are defined: shared model and parallel model. For completeness, the discussions below begin first with a review of the default (non-virtualized) model of LISP, and then cover the details of shared and parallel models.

Default (Non-Virtualized) LISP Model

By default, LISP is not virtualized in either EID space or RLOC space. That is, unless otherwise configured, both EID and RLOC addresses are resolved in the default (global) routing table. This concept is illustrated in the figure below.

Figure 35: Default (Non-Virtualized) LISP Model (Resolves Both EID and RLOC Addresses in the Default (Global) Routing Table).

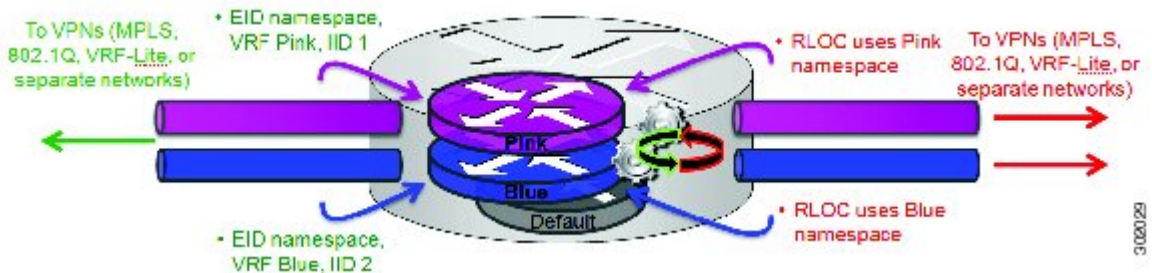


As shown in the figure above, both EID and RLOC addresses are resolved in the default table. The mapping system must also be reachable via the default table. This default model can be thought of as a single instantiation of the parallel model of LISP virtualization where EID and RLOC addresses are within the same namespace such as is the case in this default table.

LISP Parallel Model Virtualization

LISP parallel model virtualization ties virtualized EID space associated with VRFs to RLOCs associated with the same or different VRFs. This concept is illustrated in the figure below.

Figure 36: LISP parallel model virtualization resolves an EID and associated RLOCs within the same or different VRF. In this example, both EID and RLOC addresses are resolved in the same VRF, but multiple (parallel) segmentation is configured on the same device (BLUE and PINK).

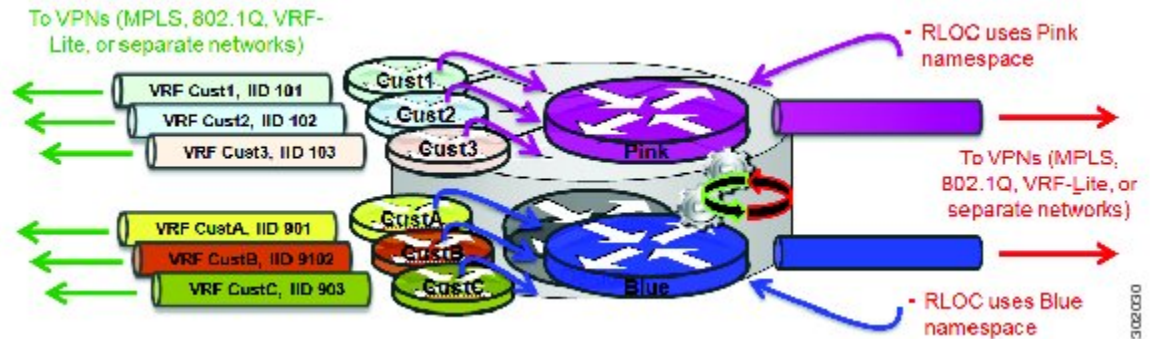


As shown in the figure above, EID space is virtualized through its association with VRFs, and these VRFs are tied to LISP Instance IDs to segment the control plane and data plane in LISP. A common, “shared” locator space, the default (global) table as shown in the figure above, is used to resolve RLOC addresses for all virtualized EIDs. The mapping system must also be reachable via the common locator space as well.

The example illustrated in the figure above shows virtualized EID space associated with a VRF (and bound to an Instance ID) being tied to locator space associated with the same VRF, in this case - Pink/Pink and Blue/Blue. However, this is not required; the EID VRF does not need to match the RLOC VRF. In any case, a mapping system must be reachable via the associated locator space. Multiple parallel instantiations can be defined.

In the most general case, shared model and parallel model may be combined such that multiple EID VRFs share a common RLOC VRF, and multiple instantiations of this architecture are implemented on the same platform, as shown in the figure below.

Figure 37: LISP shared and parallel models may be combined for maximum flexibility.

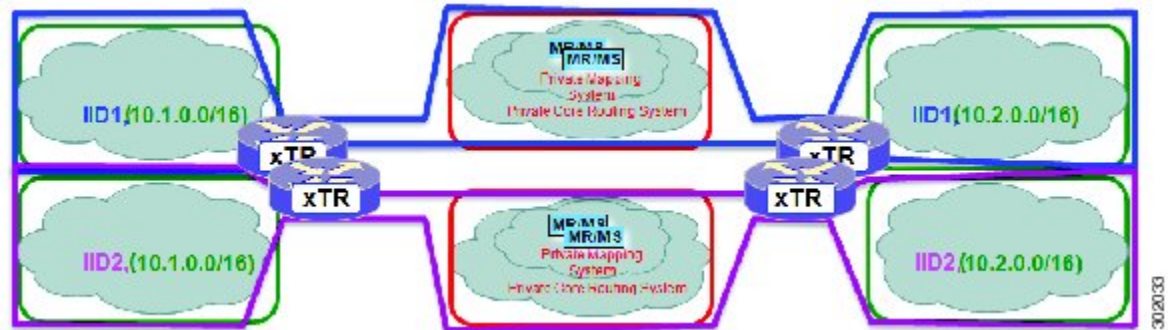


As shown in the figure above, shared and parallel models are combined to associate several EID instances to one shared RLOC VRF, and then several other EID instances to another shared RLOC VRF.

LISP Parallel Model Virtualization Architecture

Architecturally, LISP parallel model virtualization can be deployed in single or multitenancy configurations. In the parallel model multitenancy case, a set of xTRs is shared (virtualized) among multiple customers, and each customer uses their own private (segmented) core infrastructure and mapping system. All sites associated with the customer use the same instance ID and are part of a VPN using their own EID namespace as shown in the figure below.

Figure 38: In the LISP parallel model multitenancy case, shared xTRs use virtualized core networks and mapping systems. LISP instance IDs segment the LISP data plane and control plane.



LISP Parallel Model Virtualization Implementation Considerations and Caveats

When the LISP Parallel Model Virtualization is implemented, several important considerations and caveats are important. Each **router lisp** value instantiation is considered by software to be a separate process. Instance IDs must be unique only within a **router lisp** instantiation. Review the example below:

```
xTR-1(config)# vrf definition alpha
xTR-1(config-vrf)# address-family ipv4
xTR-1(config-vrf-af)# exit
xTR-1(config)# vrf definition beta
xTR-1(config-vrf)# address-family ipv4
xTR-1(config-vrf-af)# exit
xTR-1(config-vrf)# vrf definition gamma
```

```
xTR-1(config-vrf)# address-family ipv4
xTR-1(config-vrf-af)# exit
xTR-1(config-vrf)# vrf definition delta
xTR-1(config-vrf)# address-family ipv4
xTR-1(config-vrf-af)# exit
xTR-1(config-vrf)# exit
xTR-1(config)# router lisp 1
xTR-1(config-router-lisp)# locator-table vrf alpha
xTR-1(config-router-lisp)# eid-table vrf beta instance-id 101
xTR-1(config-router-lisp-eid-table)# exit
xTR-1(config-router-lisp)# exit
xTR-1(config)# router lisp 2
xTR-1(config-router-lisp)# locator-table vrf gamma
xTR-1(config-router-lisp)# eid-table vrf delta instance-id 101
xTR-1(config-router-lisp-eid-table)# exit
xTR-1(config-router-lisp)# eid-table vrf beta instance-id 201
The vrf beta table is not available for use as an EID table (in use by router lisp 1 EID
instance 101 VRF)
```

In the above example, four VRFs are created; alpha, beta, gamma, and delta. The **router lisp** instantiation **router lisp 1** is created and associated with the locator-table VRF named alpha. Next, the EID table VRF named beta is specified and associated with instance ID 101. Next, a new **router lisp** instantiation, **router lisp 2**, is created and associated with the locator-table VRF named gamma. Next, EID table VRF named delta is specified and also associated with instance ID 101. These two instance IDs are unrelated to each other; one is relevant only within **router lisp 1** and the other is only relevant within **router lisp 2**.

In the above example, also observe that while under **router lisp 2**, an attempt is made to configure an EID table VRF named beta. Note that the router is unable to use this EID table VRF since it (beta) is already associated with an **eid-table** command within the **router lisp 1** instantiation.

You can re-use an instance ID, and which EID VRF it is decapsulated into depends on the **router lisp** instantiation and locator-table VRF that it is associated with. You cannot connect the same EID VRF to more than one locator-table VRF, however.

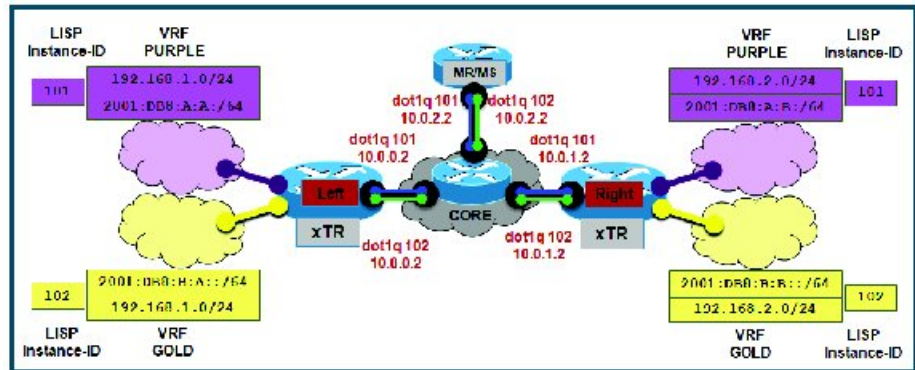
How to Configure LISP Parallel Model Virtualization

Configure Simple LISP Parallel Model Virtualization

Perform these tasks to enable and configure LISP ITR/ETR (xTR) functionality and LISP map resolver and map server for LISP parallel model virtualization.

The configuration implemented in this task and illustrated in the figure below is for two LISP sites that are connected in parallel mode. Each LISP site uses a single edge router configured as both an ITR and ETR (xTR), with a single connection to its upstream provider. However, the upstream connection is VLAN-segmented to maintain RLOC space separation within the core. Two VRFs are defined here: BLUE and GREEN. IPv4 RLOC space is used in each of these parallel networks. Both IPv4 and IPv6 EID address space is used. The LISP site registers to one map server/map resolver (MS/MR), which is segmented to maintain the parallel model architecture of the core network.

Figure 39: Simple LISP Site with One IPv4 RLOC and One IPv4 EID



The components illustrated in the topology shown in the figure above are described below:

- **LISP site:**

- The CPE functions as a LISP ITR and ETR (xTR).
- Both LISP xTRs have two VRFs: GOLD and PURPLE, with each VRF containing both IPv4 and IPv6 EID-prefixes, as shown in the figure above. Note the overlapping prefixes, used for illustration purposes. A LISP instance-id is used to maintain separation between two VRFs. Note that in this example, the share key is configured “per-VPN. ♦?”
- Each LISP xTR has a single RLOC connection to a parallel IPv4 core network.

Perform the steps in this task (once through for each xTR in the LISP site) to enable and configure LISP ITR and ETR (xTR) functionality when using a LISP map-server and map-resolver for mapping services. The example configurations at the end of this task show the full configuration for two xTRs (Left-xTR and Right-xTR).

Before you begin

The configuration below assumes that the referenced VRFs were created using the **vrf definition** command.

SUMMARY STEPS

1. **configure terminal**
2. **router lisp** *lisp-instantiation-number*
3. **locator-table vrf** *rloc-vrf-name*
4. **eid-table vrfEID-vrf-name** *instance-id instance-id*
5. **database-mapping** *EID-prefix/prefix-length locator* **priority** *priority* **weight** *weight*
6. Repeat Step 4 until all EID-to-RLOC mappings within this eid-table vrf and instance ID for this LISP site are configured.
7. **exit**
8. **ipv4 itr map-resolver** *map-resolver-address*
9. **ipv4 etr map-server** *map-server-address* **key** *key-type* *authentication-key*
10. **ipv4 itr**
11. **ipv4 etr**
12. **ipv6 itr map-resolver** *map-resolver-address*

13. `ipv6 etr map-server map-server-address key key-type authentication-key`
14. `ipv6 itr`
15. `ipv6 etr`
16. `exit`
17. `ip route vrf rloc-vrf-name ipv4-prefix next-hop`
18. `exit`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 2	router lisp lisp-instantiation-number Example: <pre>Router(config)# router lisp</pre>	Creates the specified LISP instantiation number and enters LISP configuration mode (software only). All subsequent LISP commands apply to that router LISP instantiation. <ul style="list-style-type: none"> • In this example, the router LISP instantiation 1 is configured.
Step 3	locator-table vrf rloc-vrf-name Example: <pre>Router(config-router-lisp)# locator-table vrf BLUE</pre>	Configures a router LISP instantiation to use the specified VRF as RLOC space when encapsulating EIDs and sending control plane packets. <ul style="list-style-type: none"> • In this example, the RLOC VRF named BLUE is configured.
Step 4	eid-table vrf EID-vrf-name instance-id instance-id Example: <pre>Router(config-router-lisp)# eid-table vrf PURPLE instance-id 101</pre>	Configures an association between a VRF table and a LISP instance ID, and enters eid-table configuration submode. <ul style="list-style-type: none"> • In this example, the VRF table PURPLE and instance-id 101 are associated together.
Step 5	database-mapping EID-prefix/prefix-length locator priority priority weight weight Example: <pre>Router(config-router-lisp-eid-table)# database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1</pre>	Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site. <ul style="list-style-type: none"> • In this example, a single IPv4 EID prefix, 192.168.1.0/24, within instance ID 1 at this site is associated with the local IPv4 RLOC 10.0.0.2.
Step 6	Repeat Step 4 until all EID-to-RLOC mappings within this eid-table vrf and instance ID for this LISP site are configured. Example: <pre>Router(config-router-lisp-eid-table)#</pre>	Configures an EID-to-RLOC mapping relationship and its associated traffic policy for this LISP site. <ul style="list-style-type: none"> • In this example, the IPv6 EID prefix, 2001:db8:a:a::/64, within instance ID 1 at this site is also associated with the local IPv4 RLOC 10.0.0.2.

	Command or Action	Purpose
	<pre>database-mapping 2001:db8:a:a::/64 10.0.0.2 priority 1 weight 1</pre>	
Step 7	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp-eid-table)# exit</pre>	Exits eid-table configuration submode and returns to LISP configuration mode.
Step 8	<p>ipv4 itr map-resolver map-resolver-address</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr map-resolver 10.0.2.2</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv4 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> In this example, the map resolver is specified within router lisp configuration mode. The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map resolver is reachable using its IPv4 locator address. (See the <i>LISP Command Reference Guide</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 9	<p>ipv4 etr map-server map-server-address key key-type authentication-key</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 etr map-server 10.0.2.2 key 0 PURPLE-key</pre>	<p>Configures a locator address for the LISP map server and an authentication key for which this router, acting as an IPv4 LISP ETR, will use to register with the LISP mapping system.</p> <ul style="list-style-type: none"> In this example, the map server and authentication key are specified within router lisp configuration mode. The map server must be configured with EID prefixes and instance IDs matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map-server is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 10	<p>ipv4 itr</p> <p>Example:</p> <pre>Router(config-router-lisp)# ipv4 itr</pre>	Enables LISP ITR functionality for the IPv4 address family.

	Command or Action	Purpose
Step 11	ipv4 etr Example: <pre>Router(config-router-lisp)# ipv4 etr</pre>	Enables LISP ETR functionality for the IPv4 address family.
Step 12	ipv6 itr map-resolver map-resolver-address Example: <pre>Router(config-router-lisp)# ipv6 itr map-resolver 10.0.2.2</pre>	<p>Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv6 EID-to-RLOC mapping resolutions.</p> <ul style="list-style-type: none"> In this example, the map resolver is specified within router lisp configuration mode. The locator address of the map resolver may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map-resolver is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.) <p>Note Up to two map resolvers may be configured if multiple map resolvers are available. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 13	ipv6 etr map-server map-server-address key key-type authentication-key Example: <pre>Router(config-router-lisp)# ipv6 etr map-server 10.0.2.2 key 0 PURPLE-key</pre>	<p>Configures a locator address for the LISP map-server and an authentication key that this router, acting as an IPv6 LISP ETR, will use to register to the LISP mapping system.</p> <ul style="list-style-type: none"> In this example, the map server and authentication key are specified within router lisp configuration mode. The map-server must be configured with EID prefixes and instance IDs matching those configured on this ETR and with an identical authentication key. <p>Note The locator address of the map-server may be an IPv4 or IPv6 address. In this example, because each xTR has only IPv4 RLOC connectivity, the map-server is reachable using its IPv4 locator addresses. (See the <i>LISP Command Reference Guide</i> for more details.)</p>
Step 14	ipv6 itr Example: <pre>Router(config-router-lisp)# ipv6 itr</pre>	Enables LISP ITR functionality for the IPv6 address family.
Step 15	ipv6 etr Example:	Enables LISP ETR functionality for the IPv6 address family.

	Command or Action	Purpose
	<code>Router(config-router-lisp)# ipv6 etr</code>	
Step 16	<p>exit</p> <p>Example:</p> <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 17	<p>ip route vrf rloc-vrf-name ipv4-prefix next-hop</p> <p>Example:</p> <pre>Router(config)# ip route vrf BLUE 0.0.0.0 0.0.0.0 10.0.0.1</pre>	<p>Configures a default route to the upstream next hop for all IPv4 destinations.</p> <ul style="list-style-type: none"> All IPv4 EID-sourced packets destined to both LISP and non-LISP sites are forwarded in one of two ways: <ul style="list-style-type: none"> LISP-encapsulated to a LISP site when traffic is LISP-to-LISP natively forwarded when traffic is LISP-to-non-LISP Packets are deemed to be a candidate for LISP encapsulation when they are sourced from a LISP EID and the destination matches one of the following entries: <ul style="list-style-type: none"> a current map-cache entry a default route with a legitimate next-hop no route at all <p>In this configuration example, because the xTR has IPv4 RLOC connectivity, a default route to the upstream SP is used for all IPv4 packets to support LISP processing.</p>
Step 18	<p>exit</p> <p>Example:</p> <pre>Router(config)# exit</pre>	Exits global configuration mode.

Example:

The examples below show the complete configuration for the LISP topology illustrated in the figure above and in this task. On the xTRs, the VRFs and EID prefixes are assumed to be attached to VLANs configured on the devices.

Example configuration for the Left xTR:

```
hostname Left-xTR
!
ipv6 unicast-routing
!
vrf definition PURPLE
 address-family ipv4
 exit
```

```

    address-family ipv6
    exit
  !
  vrf definition GOLD
    address-family ipv4
    exit
    address-family ipv6
    exit
  !
  interface Ethernet0/0
    ip address 10.0.0.2 255.255.255.0
  !
  interface Ethernet1/0.1
    encapsulation dot1q 101
    vrf forwarding PURPLE
    ip address 192.168.1.1 255.255.255.0
    ipv6 address 2001:DB8:A:A::1/64
  !
  interface Ethernet1/0.2
    encapsulation dot1q 102
    vrf forwarding GOLD
    ip address 192.168.1.1 255.255.255.0
    ipv6 address 2001:DB8:B:A::1/64
  !
  router lisp
    eid-table vrf PURPLE instance-id 101
      database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
      database-mapping 2001:DB8:A:A::/64 10.0.0.2 priority 1 weight 1
    eid-table vrf GOLD instance-id 102
      database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
      database-mapping 2001:DB8:B:A::/64 10.0.0.2 priority 1 weight 1
    exit
  !
  ipv4 itr map-resolver 10.0.2.2
  ipv4 itr
  ipv4 etr map-server 10.0.2.2 key Left-key
  ipv4 etr
  ipv6 itr map-resolver 10.0.2.2
  ipv6 itr
  ipv6 etr map-server 10.0.2.2 key Left-key
  ipv6 etr
  exit
  !
  ip route 0.0.0.0 0.0.0.0 10.0.0.1
  !

```

Example configuration for Right xTR:

```

hostname Right-xTR
!
ipv6 unicast-routing
!
vrf definition PURPLE
  address-family ipv4
  exit
  address-family ipv6
  exit
!
vrf definition GOLD
  address-family ipv4
  exit
  address-family ipv6
  exit
!

```

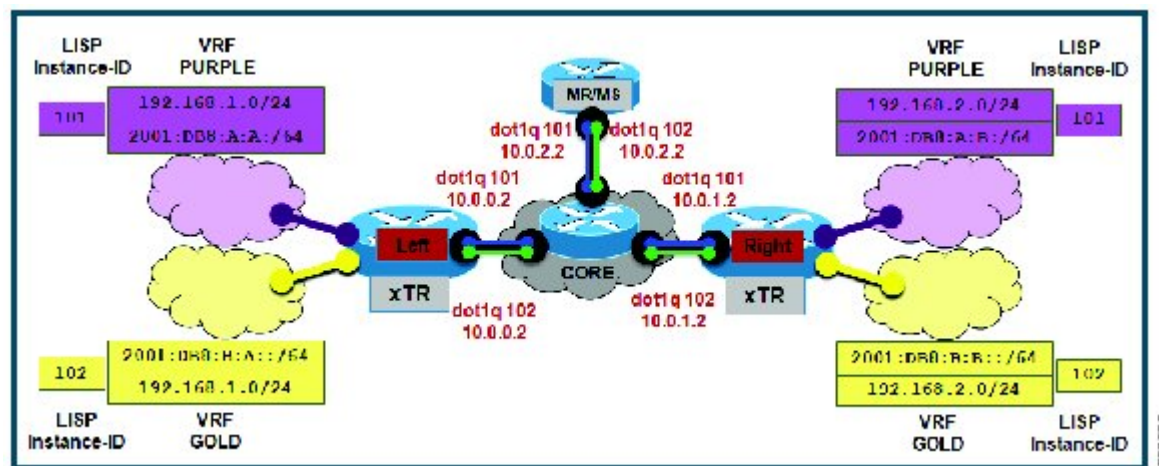
```
interface Ethernet0/0
 ip address 10.0.1.2 255.255.255.0
 !
interface Ethernet1/0.1
 encapsulation dot1q 101
 vrf forwarding PURPLE
 ip address 192.168.2.1 255.255.255.0
 ipv6 address 2001:DB8:A:B::1/64
 !
interface Ethernet1/0.2
 encapsulation dot1q 102
 vrf forwarding GOLD
 ip address 192.168.2.1 255.255.255.0
 ipv6 address 2001:DB8:B:B::1/64
 !
router lisp
 eid-table vrf PURPLE instance-id 101
  database-mapping 192.168.2.0/24 10.0.1.2 priority 1 weight 1
  database-mapping 2001:DB8:A:B::/64 10.0.1.2 priority 1 weight 1
 eid-table vrf GOLD instance-id 102
  database-mapping 192.168.2.0/24 10.0.1.2 priority 1 weight 1
  database-mapping 2001:DB8:B:B::/64 10.0.1.2 priority 1 weight 1
 exit
 !
ipv4 itr map-resolver 10.0.2.2
ipv4 itr
ipv4 etr map-server 10.0.2.2 key Right-key
ipv4 etr
ipv6 itr map-resolver 10.0.2.2
ipv6 itr
ipv6 etr map-server 10.0.2.2 key Right-key
ipv6 etr
 exit
 !
ip route 0.0.0.0 0.0.0.0 10.0.1.1
 !
```

Configuring a Private LISP Mapping System for LISP Parallel Model Virtualization

Perform this task to configure and enable standalone LISP map server/map resolver functionality for LISP parallel model virtualization. In this task, a Cisco router is configured as a standalone map resolver/map server (MR/MS) for a private LISP mapping system. Because the MR/MS is configured as a stand-alone device, it has no need for LISP alternate logical topology (ALT) connectivity. All relevant LISP sites must be configured to register with this map server so that this map server has full knowledge of all registered EID prefixes within the (assumed) private LISP system.

- **Mapping system:**

Figure 40: Simple LISP Site with One IPv4 RLOC and One IPv4 EID



- One map resolver/map server (MS/MR) system is shown in the figure above and assumed available for the LISP xTR to register to within the proper parallel RLOC space. The MS/MR has an IPv4 RLOC address of 10.0.2.2, within each VLAN/VRF (Green and Blue) providing parallel model RLOC separation in the IPv4 core.
- The map server site configurations are virtualized using LISP instance IDs to maintain separation between the two VRFs, PURPLE and GOLD.

Repeat this task for all router lisp instantiations and RLOC VRFs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router lisp** *lisp-instantiation-number*
4. **locator-table vrf** *rloc-vrf-name*
5. **site** *site-name*
6. **authentication-key** [*key-type*] *authentication-key*
7. **eid-prefix instance-id** *instance-id EID-prefix*
8. **eid-prefix instance-id** *instance-id EID-prefix*
9. **exit**
10. **ipv4 map-resolver**
11. **ipv4 map-server**
12. **ipv6 map-resolver**
13. **ipv6 map-server**
14. **exit**
15. **ip route vrf** *rloc-vrf-name ipv4-prefix next-hop*
16. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Router> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Router# configure terminal</pre>	Enters global configuration mode.
Step 3	router lisp <i>lisp-instantiation-number</i> Example: <pre>Router(config)# router lisp</pre>	Creates the specified LISP instantiation number and enters LISP configuration mode (software only). All subsequent LISP commands apply to that router LISP instantiation. <ul style="list-style-type: none"> • In this example, the router LISP instantiation 1 is configured.
Step 4	locator-table vrf <i>rloc-vrf-name</i> Example: <pre>Router(config)# locator-table vrf BLUE</pre>	Configures a router lisp instantiation to use the specified VRF as RLOC space when encapsulating EIDs and sending control plane packets. <ul style="list-style-type: none"> • In this example, the RLOC VRF BLUE is configured.
Step 5	site <i>site-name</i> Example: <pre>Router(config-router-lisp)# site Purple</pre>	Specifies a LISP site named Purple and enters LISP site configuration mode. <ul style="list-style-type: none"> • In this example, the LISP site named Purple is configured.
Step 6	authentication-key [<i>key-type</i>] <i>authentication-key</i> Example: <pre>Router(config-router-lisp-site)# authentication-key 0 Purple-key</pre>	Configures the password used to create the SHA-2 HMAC hash for authenticating the map register messages sent by an ETR when registering to the map server. <p>Note The ETR must be configured with EID prefixes and instance IDs matching the one(s) configured on this map server, as well as an identical authentication key.</p>
Step 7	eid-prefix instance-id <i>instance-id EID-prefix</i> Example: <pre>Router(config-router-lisp-site)# eid-prefix instance-id 101 192.168.1.0/24</pre>	Configures an EID prefix and instance ID that are allowed in a map register message sent by an ETR when registering to this map server. Repeat this step as necessary to configure additional IPv4 EID prefixes under this LISP site. <ul style="list-style-type: none"> • In this example, the IPv4 EID prefix 192.168.1.0/24 and instance ID 101 are associated together.

	Command or Action	Purpose
Step 8	eid-prefix instance-id <i>instance-id EID-prefix</i> Example: <pre>Router(config-router-lisp-site)# eid-prefix instance-id 101 2001:db8:a:a::/64</pre>	Configures an EID prefix and instance ID that are allowed in a map register message sent by an ETR when registering to this map server. Repeat this step as necessary to configure additional IPv6 EID prefixes under this LISP site. <ul style="list-style-type: none"> In this example, the IPv6 EID prefix 2001:db8:a:a::/64 and instance ID 101 are associated together.
Step 9	exit Example: <pre>Router(config-router-lisp-site)# exit</pre>	Exits LISP site configuration mode and returns to LISP configuration mode.
Step 10	ipv4 map-resolver Example: <pre>Router(config-router-lisp)# ipv4 map-resolver</pre>	Enables LISP map resolver functionality for EIDs in the IPv4 address family within this router lisp instantiation.
Step 11	ipv4 map-server Example: <pre>Router(config-router-lisp)# ipv4 map-server</pre>	Enables LISP map server functionality for EIDs in the IPv4 address family within this router lisp instantiation.
Step 12	ipv6 map-resolver Example: <pre>Router(config-router-lisp)# ipv6 map-resolver</pre>	Enables LISP map resolver functionality for EIDs in the IPv6 address family within this router lisp instantiation.
Step 13	ipv6 map-server Example: <pre>Router(config-router-lisp)# ipv6 map-server</pre>	Enables LISP map server functionality for EIDs in the IPv6 address family within this router lisp instantiation.
Step 14	exit Example: <pre>Router(config-router-lisp)# exit</pre>	Exits LISP configuration mode and returns to global configuration mode.
Step 15	ip route vrf <i>rloc-vrf-name ipv4-prefix next-hop</i> Example: <pre>Router(config)# ip route vrf BLUE 0.0.0.0 0.0.0.0 10.0.2.1</pre>	Configures a default route to the upstream next hop for all IPv4 destinations, reachable within the specified RLOC VRF.
Step 16	exit Example:	Exits global configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
	Router(config)# exit	

Example:

Example configuration for the map server/map resolver.

```

hostname MSMR
!
vrf definition BLUE
  address-family ipv4
  exit
!
vrf definition GREEN
  address-family ipv4
  exit
!
ipv6 unicast-routing
!
interface Ethernet0/0.101
  encapsulation dot1Q 101
  vrf forwarding BLUE
  ip address 10.0.0.2 255.255.255.0
!
interface Ethernet0/0.102
  encapsulation dot1Q 102
  vrf forwarding GREEN
  ip address 10.0.0.2 255.255.255.0
!
router lisp 1
  locator-table vrf BLUE
  site Purple
    authentication-key PURPLE-key
    eid-prefix instance-id 101 192.168.1.0/24
    eid-prefix instance-id 101 192.168.2.0/24
    eid-prefix instance-id 101 2001:DB8:A:A::/64
    eid-prefix instance-id 101 2001:DB8:A:B::/64
  !
  ipv4 map-server
  ipv4 map-resolver
  ipv6 map-server
  ipv6 map-resolver
!
router lisp 2
  locator-table vrf GREEN
  site Gold
    authentication-key GOLD-key
    eid-prefix instance-id 102 192.168.1.0/24
    eid-prefix instance-id 102 192.168.2.0/24
    eid-prefix instance-id 102 2001:DB8:B:A::/64
    eid-prefix instance-id 102 2001:DB8:B:B::/64
  !
  ipv4 map-server
  ipv4 map-resolver
  ipv6 map-server
  ipv6 map-resolver
!
ip route vrf GREEN 0.0.0.0 0.0.0.0 10.0.2.1
ip route vrf BLUE 0.0.0.0 0.0.0.0 10.0.2.1

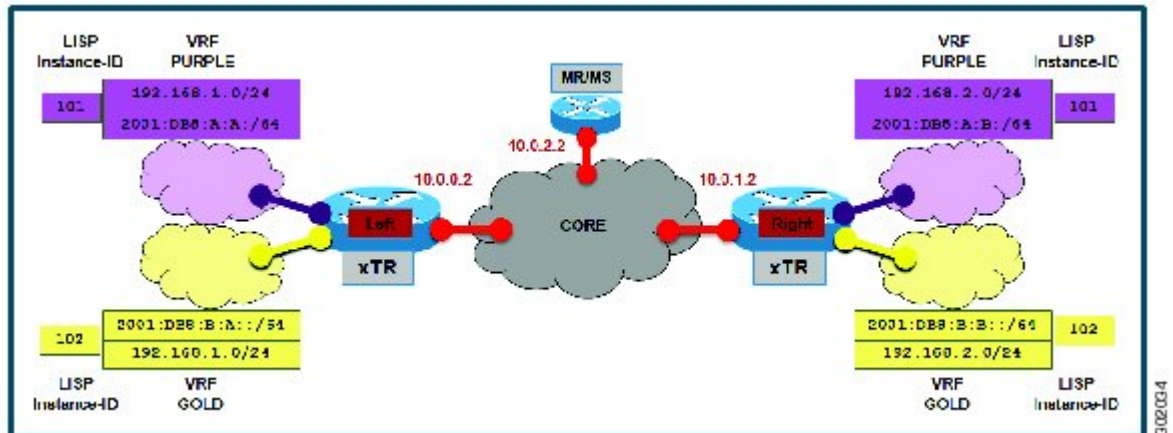
```

Verifying and Troubleshooting LISP Virtualization

After configuring LISP, verifying and troubleshooting LISP configuration and operations may be performed by following the optional steps described below. Note that certain verification and troubleshooting steps may only apply to certain types of LISP devices.

In this task, the topology is shown in the figure below and the configuration is from the “Configure Simple LISP Shared Model Virtualization” task, but the commands are applicable to both LISP shared and parallel model virtualization.

Figure 41: Simple LISP Site with Virtualized IPv4 and IPv6 EIDs and a Shared IPv4 Core



Note

The following examples do not show every available command and every available output display. Refer to the *Cisco IOS LISP Command Reference* for detailed explanations of each command.

SUMMARY STEPS

1. **enable**
2. **show running-config | section router lisp**
3. **show [ip | ipv6] lisp**
4. **show [ip | ipv6] lisp map-cache**
5. **show [ip | ipv6] lisp database [eid-table vrf vrf-name]**
6. **show lisp site [name site-name]**
7. **lig {[self {ipv4 | ipv6}] | {hostname | destination-EID}}**
8. **ping {hostname | destination-EID}**
9. **clear [ip | ipv6] lisp map-cache**

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 show running-config | section router lisp

The **show running-config | section router lisp** command is useful for quickly verifying the LISP configuration on the device. This command applies to any LISP device. The following is sample output from the **show running-config | section router lisp** command when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and a shared IPv4 core:

Example:

```
Router# show running-config | section router lisp

router lisp
  eid-table vrf PURPLE instance-id 101
    database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
    database-mapping 2001:DB8:A:A::/64 10.0.0.2 priority 1 weight 1
  eid-table vrf GOLD instance-id 102
    database-mapping 192.168.1.0/24 10.0.0.2 priority 1 weight 1
    database-mapping 2001:DB8:B:A::/64 10.0.0.2 priority 1 weight 1
  exit
!
  ipv4 itr map-resolver 10.0.2.2
  ipv4 itr
  ipv4 etr map-server 10.0.2.2 key Left-key
  ipv4 etr
  ipv6 itr map-resolver 10.0.2.2
  ipv6 itr
  ipv6 etr map-server 10.0.2.2 key Left-key
  ipv6 etr
  exit
```

Step 3 show [ip | ipv6] lisp

The **show ip lisp** and **show ipv6 lisp** commands are useful for quickly verifying the operational status of LISP as configured on the device, as applicable to the IPv4 and IPv6 address families respectively. This command applies to any LISP device.

Example:

The first example shows a summary of LISP operational status and IPv6 address family information by EID table:

```
Router# show ipv6 lisp eid-table summary

Instance count: 2
Key: DB - Local EID Database entry count (@ - RLOC check pending
      * - RLOC consistency problem),
     DB no route - Local EID DB entries with no matching RIB route,
     Cache - Remote EID mapping cache size, IID - Instance ID,
     Role - Configured Role

EID VRF name      Interface      DB  DB no  Cache Incom  Cache
                  (.IID)  size  route  size plete  Idle Role
PURPLE            LISP0.101     1    0     1  0.0%  0.0% ITR-ETR
GOLD              LISP0.102     1    0     1  0.0%  0.0% ITR-ETR
```

Example:

The second example shows LISP operational status and IPv6 address family information for the VRF named PURPLE:

```
Router# show ipv6 lisp eid-table vrf PURPLE
```

```

Instance ID:                101
Router-lisp ID:             0
Locator table:              default
EID table:                  PURPLE
Ingress Tunnel Router (ITR): enabled
Egress Tunnel Router (ETR): enabled
Proxy-ITR Router (PITR):   disabled
Proxy-ETR Router (PETR):   disabled
Map Server (MS):           disabled
Map Resolver (MR):         disabled
Map-Request source:        2001:DB8:A:A::1
ITR Map-Resolver(s):       10.0.2.2
ETR Map-Server(s):         10.0.2.2 (00:00:24)
ITR use proxy ETR RLOC(s): none

```

Example:

The third example shows LISP operational status and IPv6 address family information for the instance ID of 101:

```

Router# show ipv6 lisp instance-id 101

Instance ID:                101
Ingress Tunnel Router (ITR): enabled
Egress Tunnel Router (ETR): enabled
Proxy-ITR Router (PITR):   disabled
Proxy-ETR Router (PETR):   disabled
Map Server (MS):           disabled
Map Resolver (MR):         disabled
Map-Request source:        2001:DB8:A:A::1
ITR Map-Resolver(s):       10.0.2.2
ETR Map-Server(s):         10.0.2.2 (00:00:11)
ITR Solicit Map Request (SMR): accept and process
  Max SMRs per map-cache entry: 8 more specifics
  Multiple SMR suppression time: 60 secs
ETR accept mapping data:   disabled, verify disabled
ETR map-cache TTL:         1d00h

```

Step 4 `show [ip | ipv6] lisp map-cache`

The `show ip lisp map-cache` and `show ipv6 lisp map-cache` commands are useful for quickly verifying the operational status of the map cache on a device configured as an ITR or PITR, as applicable to the IPv4 and IPv6 address families respectively.

Example:

The following example shows IPv6 mapping cache information based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and a shared IPv4 core. This example output assumes that a map-cache entry has been received for another site with the IPv6 EID prefix 2001:db8:b:b::/64.

```

Router# show ip lisp map-cache eid-table vrf GOLD

LISP IPv6 Mapping Cache for EID-table vrf GOLD (IID 102), 2 entries

::/0, uptime: 01:09:52, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
2001:DB8:B:B::/64, uptime: 00:00:10, expires: 23:59:42, via map-reply, complete
Locator  Uptime  State  Pri/Wgt
10.0.1.2 00:00:10 up      1/1

```

Step 5 `show [ip | ipv6] lisp database [eid-table vrf vrf-name]`

The **show ip lisp database** and **show ipv6 lisp database** commands are useful for quickly verifying the operational status of the database mapping on a device configured as an ETR, as applicable to the IPv4 and IPv6 address families respectively.

Example:

The following example shows IPv6 mapping database information for the VRF named GOLD.

```
Router# show ipv6 lisp database eid-table vrf GOLD

LISP ETR IPv6 Mapping Database for EID-table vrf GOLD (IID 102), LSBs: 0x1, 1 entries

EID-prefix: 2001:DB8:B:A::/64
  10.0.0.2, priority: 1, weight: 1, state: site-self, reachable
```

Step 6 **show lisp site [name site-name]**

The **show lisp site** command is useful for quickly verifying the operational status of LISP sites, as configured on a map server. This command only applies to a device configured as a map server. The following example output is based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and shows the information for the instance ID of 101.

Example:

```
Router# show lisp site instance-id 101

LISP Site Registration Information

Site Name      Last      Up    Who Last      Inst  EID Prefix
              Register  Registered
Left           00:00:36 yes  10.0.0.2     101   192.168.1.0/24
              00:00:43 yes  10.0.0.2     101   2001:DB8:A:A::/64
Right          00:00:31 yes  10.0.1.2     101   192.168.2.0/24
              00:00:02 yes  10.0.1.2     101   2001:DB8:A:B::/64
```

Example:

This second example shows LISP site information for the IPv6 EID prefix of 2001:db8:a:a:/64 and instance ID of 101.

```
Router# show lisp site 2001:db8:a:a:/64 instance-id 101

LISP Site Registration Information

Site name: Left
Allowed configured locators: any
Requested EID-prefix:
  EID-prefix: 2001:DB8:A:A::/64 instance-id 101
  First registered: 02:41:55
  Routing table tag: 0
  Origin: Configuration
Registration errors:
  Authentication failures: 4
  Allowed locators mismatch: 0
ETR 10.0.0.2, last registered 00:00:22, no proxy-reply, no map-notify
  TTL 1d00h
  Locator Local State Pri/Wgt
  10.0.0.2 yes up 1/1
```

Step 7 **lig {[self {ipv4 | ipv6}] | {hostname | destination-EID}}**

The LISP Internet Groper (lig) command is useful for testing the LISP control plane. The **lig** command can be used to query for the indicated destination hostname or EID, or the routers local EID-prefix. This command provides a simple means of testing whether a destination EID exists in the LISP mapping database system, or your site is registered with

the mapping database system. This command is applicable for both the IPv4 and IPv6 address families and applies to any LISP device that maintains a map cache (for example, if configured as an ITR or PITR). The following example output is based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes and shows the information for the instance ID of 101 and the IPv4 EID prefix of 192.168.2.1.

Example:

```
Router# lig instance-id 101 192.168.2.1

Mapping information for EID 192.168.2.1 from 10.0.1.2 with RTT 12 msecs
192.168.2.0/24, uptime: 00:00:00, expires: 23:59:52, via map-reply, complete
  Locator  Uptime    State      Pri/Wgt
  10.0.1.2 00:00:00  up         1/1
```

Example:

This second example output shows information about the VRF named PURPLE:

```
Router# lig eid-table vrf PURPLE self

Mapping information for EID 192.168.1.0 from 10.0.0.1 with RTT 20 msecs
192.168.1.0/24, uptime: 00:00:00, expires: 23:59:52, via map-reply, self
  Locator  Uptime    State      Pri/Wgt
  10.0.0.1 00:00:00  up, self   1/1
```

Step 8 ping {hostname | destination-EID}

The **ping** command is useful for testing basic network connectivity and reachability and/or liveness of a destination EID or RLOC address. When using **ping** it is important to be aware that because LISP uses an encapsulation, you should always specify a source address; never allow the **ping** application to assign its own default source address. This is because there are four possible ways to use **ping**, and without explicitly indicating the source address, the wrong one may be used by the application leading to erroneous results that complicate operational verification or troubleshooting. The four possible uses of **ping** include:

- RLOC-to-RLOC—Sends “echo” packets out natively (no LISP encap) and receive the “echo-reply” back natively. This can be used to test the underlying network connectivity between locators of various devices, such as xTR to Map-Server or Map-Resolver.
- EID-to-EID—Sends “echo” packets out LISP-encaped and receive the “echo-reply” back LISP-encaped. This can be used to test the LISP data plane (encapsulation) between LISP sites.
- EID-to-RLOC—Sends “echo” packets out natively (no LISP encap) and receive the “echo-reply” back LISP-encaped through a PITR mechanism. This can be used to test the PITR infrastructure.
- RLOC-to-EID - Sends “echo” packets out LISP-encaped and receive the “echo-reply” back natively. This can be used to test PETR capabilities.

The **ping** command is applicable to the IPv4 and IPv6 address families respectively, and can be used on any LISP device in some manner. (The ability to do LISP encapsulation, for example, requires the device to be configured as an ITR or PITR.)

The following example output from the **ping** command is based on a configuration when a simple LISP site is configured with virtualized IPv4 and IPv6 EID prefixes. (Note that ping is not a LISP command and does not know about an EID table or an instance ID. When virtualization is included, output limiters can only be specified by VRF.)

Example:

```
Router# ping vrf PURPLE 2001:DB8:a:b::1 source 2001:DB8:a:a::1 rep 100
```



```
Type escape sequence to abort.
Sending 100, 100-byte ICMP Echos to 2001:DB8:A:B::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:A:A::1%PURPLE
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (100/100), round-trip min/avg/max = 0/0/1 ms
```

Example:

```
Router# ping vrf GOLD

Protocol [ip]: ipv6
Target IPv6 address: 2001:db8:b:b::1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands? [no]: y
Source address or interface: 2001:db8:b:a::1
.
.
.
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:B:B::1, timeout is 2 seconds:
Packet sent with a source address of 2001:DB8:B:A::1%GOLD
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
```

Step 9 clear [ip | ipv6] lisp map-cache

The **clear ip lisp map-cache** and **clear ipv6 lisp map-cache** commands remove all IPv4 or IPv6 dynamic LISP map-cache entries stored by the router. This can be useful trying to quickly verify the operational status of the LISP control plane. This command applies to a LISP device that maintains a map cache (for example, if configured as an ITR or PITR).

Example:

The following example displays IPv4 mapping cache information for instance ID 101, shows the command used to clear the mapping cache for instance ID 101, and displays the show information after clearing the cache.

```
Router# show ip lisp map-cache instance-id 101

LISP IPv4 Mapping Cache for EID-table vrf PURPLE (IID 101), 2 entries

0.0.0.0/0, uptime: 00:25:17, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
192.168.2.0/24, uptime: 00:20:13, expires: 23:39:39, via map-reply, complete
  Locator   Uptime    State     Pri/Wgt
  10.0.1.2   00:20:13  up        1/1

Router# clear ip lisp map-cache instance-id 101

Router# show ip lisp map-cache instance-id 101

LISP IPv4 Mapping Cache, 1 entries

0.0.0.0/0, uptime: 00:00:02, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
```

Configuration Examples for LISP Parallel Model Virtualization

Complete configuration examples are available within each task under the “How to Configure LISP Parallel Model Virtualization” section.

Additional References

Related Documents

Document Title	Location
Cisco IOS IP Routing: LISP Command Reference	http://www.cisco.com/en/US/docs/ios-xml/ios/iproute_lisp/command/ip-lisp-cr-book.html
Enterprise IPv6 Transitions Strategy Using the Locator/ID Separation Protocol	Cisco LISP Software Image Download Page
Cisco IOS LISP0 Virtual Interface, Application Note, Version 1.0	Cisco LISP Software Image Download Page
Cross-Platform Release Notes for Cisco IOS Release 15.2M&T	http://www.cisco.com/en/US/docs/ios/15_2m_and_t/release/notes/15_2m_and_t.html

Standards

Standard	Title
IANA Address Family Numbers	http://www.iana.org/assignments/address-family-numbers/address-family-numbers.xml

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
draft-ietf-lisp-22	Locator/ID Separation Protocol (LISP) http://tools.ietf.org/html/draft-ietf-lisp-22
draft-ietf-lisp-ms-16	LISP Map Server http://tools.ietf.org/html/draft-ietf-lisp-ms-16
draft-ietf-lisp-alt-10	LISP Alternative Topology (LISP+ALT) http://tools.ietf.org/html/draft-ietf-lisp-alt-10
draft-ietf-lisp-LCAF-06	LISP Canonical Address Format (LCAF) http://tools.ietf.org/wg/lisp/
draft-ietf-lisp-interworking-06	Interworking LISP with IPv4 and IPv6 http://tools.ietf.org/html/draft-ietf-lisp-interworking-06
draft-ietf-lisp-lig-06	LISP Internet Groper (LIG) http://tools.ietf.org/html/draft-ietf-lisp-lig-06
draft-ietf-lisp-mib-03	LISP MIB http://tools.ietf.org/wg/lisp/draft-ietf-lisp-mib/

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP Parallel Model Virtualization

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 5: Feature Information for LISP Parallel Model Virtualization

Feature Name	Releases	Feature Information
LISP Parallel Model Virtualization	15.2(3)T	LISP Parallel Model Virtualization ties virtualized EID space associated with VRFs to RLOCs associated with the same or different VRFs.



CHAPTER 7

LISP Host Mobility Across Subnet

- [Finding Feature Information, on page 171](#)
- [Information About LISP Host Mobility Across Subnet, on page 171](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About LISP Host Mobility Across Subnet

Devices configured with LISP Host Mobility ASM have the following characteristics:

- Each edge router (xTR) is the first Layer-3 hop
- Proxy-arp is enabled on the xTR's gateway interface
- Each roaming site xTR should register with a common set of map-servers
- Mobility hosts should not be "silent" after they move
- A multicast configuration is needed by xTRs only if the site has multiple xTRs, for example for HSRP. A single xTR does not need to use multicasting.
- Supports vmotion or live host mobility only in the case of North-South traffic
- LISP encapsulation (ASM) is required for East-West traffic.

Overview of LISP Host Mobility Across Subnet

You can use LISP Host Mobility Across Subnet commands to deploy extended subnets and across subnets. A detailed configuration guide and examples are under development and will appear here soon. Meanwhile, please refer to the LISP Command Reference.



CHAPTER 8

LISP Delegate Database Tree (DDT)

- [Finding Feature Information](#), on page 173
- [Information About Delegate Database Tree \(DDT\)](#), on page 173

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Delegate Database Tree (DDT)

Overview of LISP Delegate Database Tree (DDT)

You can use LISP Delegate Database Tree (DDT) commands to deploy a distributed LISP Mapping System. A detailed configuration guide and examples are under development and will appear here soon. Meanwhile, please refer to the *LISP Command Reference*.



CHAPTER 9

LISP ESM Multihop Mobility

The LISP ESM Multihop Mobility feature separates the Locator/ID Separation Protocol (LISP) dynamic host detection function from the LISP encapsulation/decapsulation function within a LISP topology.

- [Finding Feature Information, on page 175](#)
- [Restrictions for LISP ESM Multihop Mobility, on page 175](#)
- [Information About LISP ESM Multihop Mobility, on page 176](#)
- [How to Configure LISP ESM Multihop Mobility, on page 178](#)
- [Configuration Examples for LISP ESM Multihop Mobility, on page 188](#)
- [Additional References for LISP ESM Multihop Mobility, on page 190](#)
- [Feature Information for LISP ESM Multihop Mobility, on page 190](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for LISP ESM Multihop Mobility

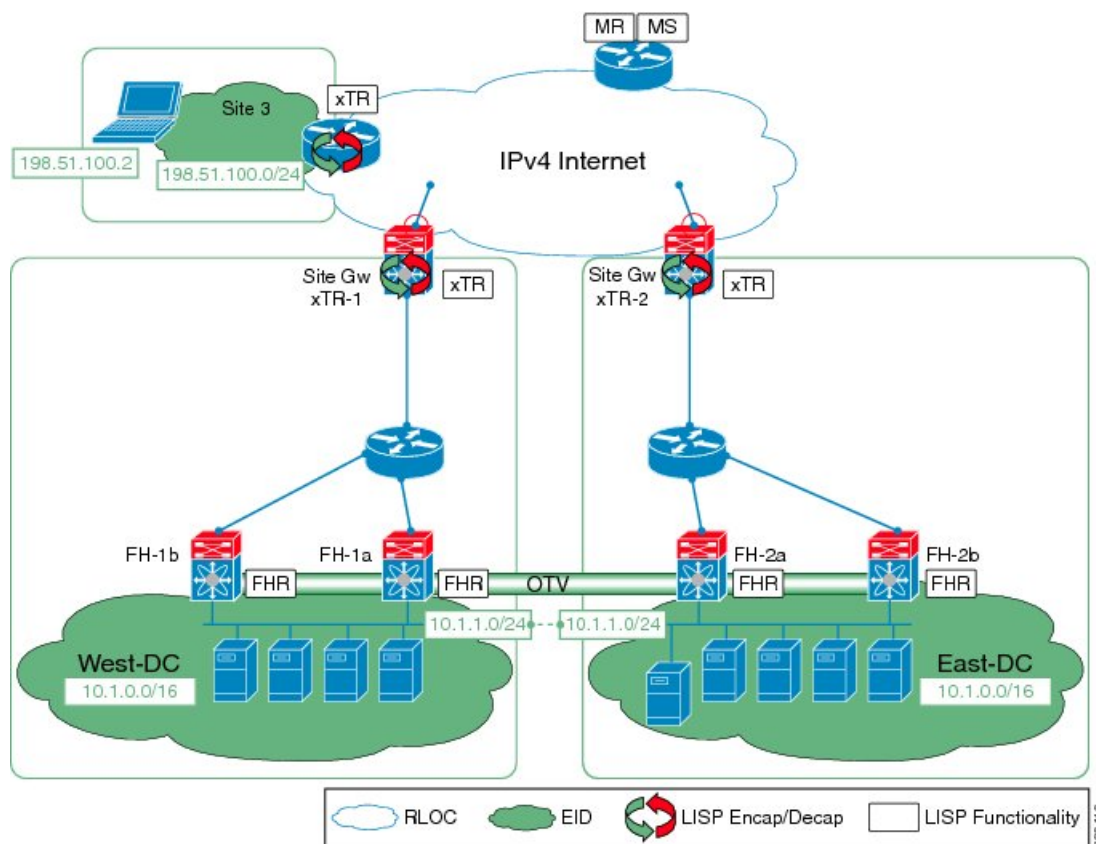
- Supports Locator/ID Separation Protocol (LISP) multihop mobility only in Extended Subnet Mode (ESM) with Overlay Transport Virtualization (OTV).
- Requires OTV First Hop Redundancy Protocol (FHRP) isolation to avoid hair-pinning of traffic across the OTV Data Center Interconnect (DCI) framework.
- Does not support Network Address Translated (NAT'd) endpoint identifiers (EIDs).

Information About LISP ESM Multihop Mobility

LISP ESM Multihop Mobility Overview

A first-hop router (FHR) detects the presence of a dynamic host endpoint identifier (EID) and notifies the site gateway xTR. A device configured as both an ingress tunnel router (ITR) and an egress tunnel router (ETR) is known as an xTR. The site gateway xTR registers the dynamic EID with a map server. The Site Gateway xTR performs Locator/ID Separation Protocol (LISP) encapsulation/decapsulation of the traffic from or to the dynamic EID to or from remote sites.

Figure 42: LISP ESM Multihop Mobility Sample Topology



Multiple Layer 3 hops can exist between the FHR and the site gateway xTR when deploying the LISP ESM Multihop Mobility feature. You can insert non-LISP devices like firewalls and load-balancers into the data center.



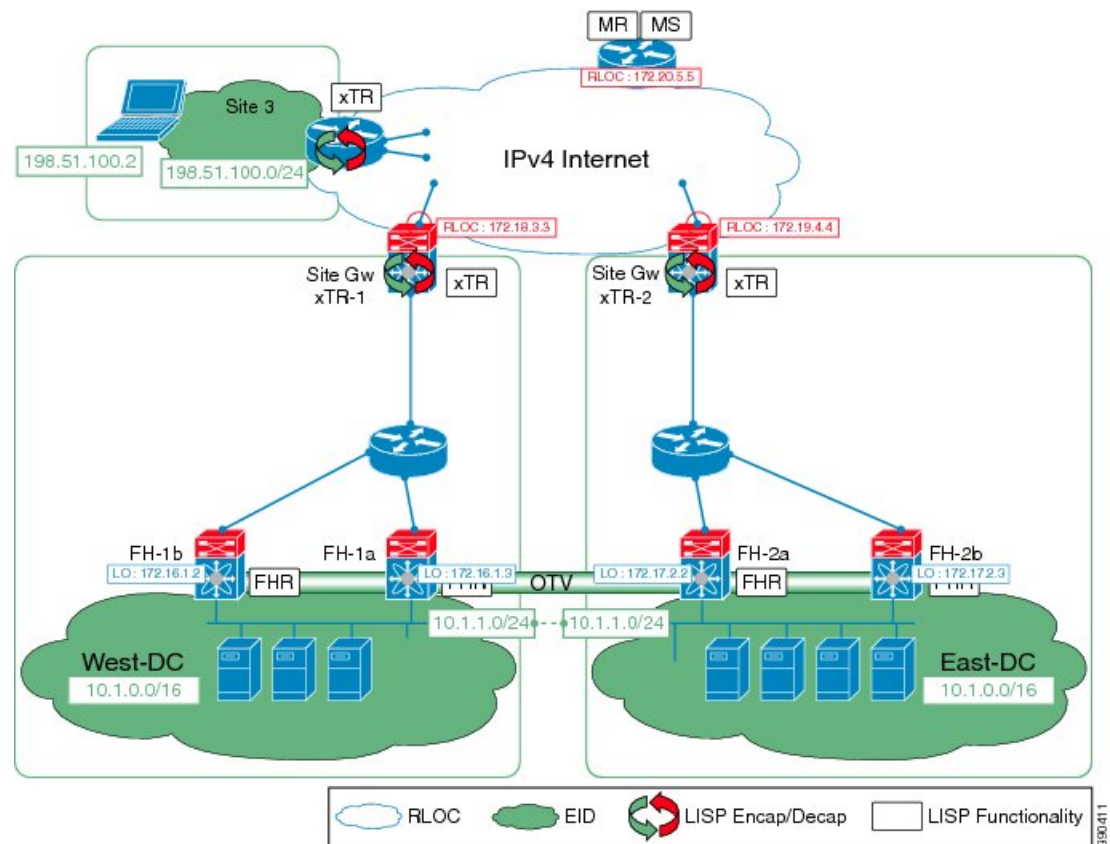
Note LISP supports silent host moves from the 15.4(1)T release.



Note LISP supports redistributing host routes for servers discovered by LISP into Interior Gateway Protocol (IGP) via Open Shortest Path First (OSPF) protocol/ Intermediate System-to-Intermediate System (IS-IS) protocol/ Routing Information Protocol (RIP)/ Border Gateway Protocol (BGP).

Perform the tasks shown below to configure LISP ESM multihop mobility on a Locator ID/Separation Protocol (LISP) site with three IPv4 routing locators (RLOCs). In these tasks, a LISP site uses a single edge router configured as both an ITR and an ETR (known as an xTR) with two connections to the upstream provider. Both the RLOCs and the endpoint identifier (EID) prefix are IPv4. The LISP site registers to a map resolver map server (MRMS) device in the network core. The topology used in this LISP configuration is shown in the figure below.

Figure 43: Topology for LISP ESM Multihop Mobility



The components illustrated in the topology shown in the above figure are described below:

LISP Site

- The customer premises equipment (CPE) functions as a LISP ITR and ETR (xTR).
- The LISP xTR is authoritative for the IPv4 EID prefix of 10.1.0.0/16.
- The LISP xTR has two RLOC connections to the core. The RLOC connection to xTR-1 is 172.18.3.3; the RLOC connection to xTR-2 is 172.19.4.4.

Mapping System

- An MRMS system is assumed to be available for the LISP xTRs to configure. The MRMS has IPv4 RLOCs 10.1.1.0 and 10.1.1.9.
- Mapping services are assumed to be provided as part of this LISP solution via a private mapping system or as a public LISP mapping system.

How to Configure LISP ESM Multihop Mobility

Configuring First-Hop Router

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router lisp**
4. **locator-set** *locator-set-name*
5. *ipv4-address* **priority** *priority-locator* **weight** *locator-weight*
6. Repeat Step 5 to configure another locator entry.
7. **exit**
8. **eid-table default instance-id** *id*
9. **dynamic-eid** *dynamic-eid-name*
10. **database-mapping** *dynamic-eid-prefix/prefix-length* **locator-set** *name*
11. **eid-notify** *ipv4-address* **key** *password*
12. **map-notify-group** *ipv4-group-address*
13. **exit**
14. **exit**
15. **exit**
16. **interface** *type number*
17. **lisp mobility** *dynamic-eid-name*
18. **lisp extended-subnet-mode**
19. **ip address** *ip-address mask*
20. **standby** *group-number* **ip** *virtual-ip-address*
21. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	router lisp Example: Device(config)# router lisp	Enters LISP configuration mode.
Step 4	locator-set locator-set-name Example: Device(config-router-lisp)# locator-set WestDC	Specifies a locator set and enters LISP locator-set configuration mode.
Step 5	ipv4-address priority priority-locator weight locator-weight Example: Device(config-router-lisp-locator-set)# 172.16.1.2 priority 10 weight 50	Configures the LISP locator set. The LISP locator set is the set of addresses that the first-hop router (FHR) uses while communicating with the gateway xTR. You can configure each locator address by creating a locator entry with an assigned priority and weight.
Step 6	Repeat Step 5 to configure another locator entry.	—
Step 7	exit Example: Device(config-router-lisp-locator-set)# exit	Exits LISP locator-set configuration mode and returns to LISP configuration mode.
Step 8	eid-table default instance-id id Example: Device(config-router-lisp)# eid-table default instance-id 0	Configures an association between the default virtual routing and forwarding (VRF) table and a LISP instance ID, and enters EID table configuration mode.
Step 9	dynamic-eid dynamic-eid-name Example: Device(config-router-lisp-eid-table)# dynamic-eid VMs	Specifies a LISP virtual machine (VM)-mobility (dynamic EID roaming) policy and enters dynamic EID configuration mode.
Step 10	database-mapping dynamic-eid-prefix/prefix-length locator-set name Example: Device(config-router-lisp-eid-table-dynamic-eid)# database-mapping 10.1.1.0/24 locator-set WestDC	Configures an IPv4 mapping relationship and an associated traffic policy for the LISP VM-mobility (dynamic EID) policy. Note You can enter the limit dynamic value keyword to limit the number of discoverable dynamic EIDs. However, if you have enabled debug mode (using the service internal command), then the number of discoverable dynamic EIDs will be increased to a fixed value of 65535.
Step 11	eid-notify ipv4-address key password Example: Device(config-router-lisp-eid-table-dynamic-eid)# eid-notify 192.0.2.21 key k	Enables sending of dynamic endpoint identifier (EID) presence notifications to a gateway xTR with the specified IPv4 address along with the authentication key used with the gateway xTR.

	Command or Action	Purpose
Step 12	map-notify-group <i>ipv4-group-address</i> Example: Device(config-router-lisp-eid-table-dynamic-eid) # map-notify-group 224.0.0.0	Specifies the IPv4 multicast group address used for sending and receiving site-based map-notify multicast messages.
Step 13	exit Example: Device(config-router-lisp-eid-table-dynamic-eid) # exit	Exits dynamic EID configuration mode and returns to EID table configuration mode.
Step 14	exit Example: Device(config-router-lisp-eid-table) # exit	Exits EID table configuration mode and returns to LISP configuration mode.
Step 15	exit Example: Device(config-router-lisp) # exit	Exits LISP configuration mode and returns to global configuration mode.
Step 16	interface <i>type number</i> Example: Device(config) # interface Vlan 11	Specifies the interface type and number and enters interface configuration mode.
Step 17	lisp mobility <i>dynamic-eid-name</i> Example: Device(config-if) # lisp mobility VMs	Allows EID mobility on the interface and specifies the name of the dynamic EID.
Step 18	lisp extended-subnet-mode Example: Device(config-if) # lisp extended-subnet-mode	Enables extended subnet mode on the interface.
Step 19	ip address <i>ip-address mask</i> Example: Device(config-if) # ip address 10.1.1.2 255.255.255.0	Configures an IPv4 address for a specific interface.
Step 20	standby <i>group-number ip virtual-ip-address</i> Example: Device(config-if) # standby 1 ip 10.1.1.1	Enables IPv4 Hot Standby Router Protocol (HSRP) and sets the virtual IP address.
Step 21	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Configuring Site Gateway xTR

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router lisp**
4. **locator-set** *locator-set-name*
5. *ipv4-address* **priority** *priority-locator* **weight** *locator-weight*
6. **exit**
7. **eid-table default instance-id** *id*
8. **database-mapping** *dynamic-eid-prefix/prefix-length* **locator-set** *name*
9. **dynamic-eid** *dynamic-eid-name*
10. **database-mapping** *dynamic-eid-prefix/prefix-length* **locator-set** *name*
11. **eid-notify authentication-key** *password*
12. **exit**
13. **exit**
14. **ipv4 itr map-resolver** *map-resolver-address*
15. **ipv4 itr**
16. **ipv4 etr map-server** *map-server-address* **key** *authentication-key*
17. **ipv4 etr**
18. **exit**
19. **interface** *type number*
20. **ip address** *ip-address mask*
21. **lisp mobility** *dynamic-eid-name*
22. **lisp extended-subnet-mode**
23. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router lisp Example: Device(config)# router lisp	Enters LISP configuration mode.
Step 4	locator-set <i>locator-set-name</i> Example:	Specifies a locator set and enters LISP locator-set configuration mode.

	Command or Action	Purpose
	Device(config-router-lisp)# locator-set WestDC	
Step 5	<p>ipv4-address priority priority-locator weight locator-weight</p> <p>Example:</p> <pre>Device(config-router-lisp-locator-set)# 172.18.3.3 priority 10 weight 50</pre>	Configures the LISP locator set. The LISP locator set is the set of addresses used by the gateway xTR while encapsulating/decapsulating LISP traffic from and to the endpoint identifier (EID).
Step 6	<p>exit</p> <p>Example:</p> <pre>Device(config-router-lisp-locator-set)# exit</pre>	Exits LISP locator-set configuration mode and returns to LISP configuration mode.
Step 7	<p>eid-table default instance-id id</p> <p>Example:</p> <pre>Device(config-router-lisp)# eid-table default instance-id 0</pre>	Configures an association between the default virtual routing and forwarding (VRF) table and a LISP instance ID, and enters EID table configuration mode.
Step 8	<p>database-mapping dynamic-eid-prefix/prefix-length locator-set name</p> <p>Example:</p> <pre>Device(config-router-lisp-eid-table)# database-mapping 10.1.0.0/16 locator-set WestDC</pre>	<p>Configures an IPv4 mapping relationship and an associated traffic policy for LISP virtual machine (VM)-mobility (dynamic EID) policy.</p> <p>Note You can enter the limit dynamic value keyword to limit the number of discoverable dynamic EIDs. However, if you have enabled debug mode (using the service internal command), then the number of discoverable dynamic EIDs will be increased to a fixed value of 65535.</p>
Step 9	<p>dynamic-eid dynamic-eid-name</p> <p>Example:</p> <pre>Device(config-router-lisp-eid-table)# dynamic-eid VMs</pre>	Specifies a LISP VM-mobility (dynamic EID roaming) policy and enters dynamic EID configuration mode.
Step 10	<p>database-mapping dynamic-eid-prefix/prefix-length locator-set name</p> <p>Example:</p> <pre>Device(config-router-lisp-eid-table-dynamic-eid)# database-mapping 10.1.1.0/24 locator-set WestDC</pre>	Configures an IPv4 mapping relationship and an associated traffic policy for LISP VM-mobility (dynamic EID) policy.
Step 11	<p>eid-notify authentication-key password</p> <p>Example:</p> <pre>Device(config-router-lisp-eid-table-dynamic-eid)# eid-notify authentication-key k</pre>	Specifies the authentication key to validate the EID-notify sent from a first-hop router (FHR).
Step 12	<p>exit</p> <p>Example:</p>	Exits dynamic EID configuration mode and returns to EID table configuration mode.

	Command or Action	Purpose
	Device(config-router-lisp-eid-table-dynamic-eid)# exit	
Step 13	exit Example: Device(config-router-lisp-eid-table)# exit	Exits EID table configuration mode and returns to LISP configuration mode.
Step 14	ipv4 itr map-resolver map-resolver-address Example: Device(config-router-lisp)# ipv4 itr map-resolver 172.20.5.5	Configures a locator address for the LISP map resolver to which this device will send map request messages for IPv4 EID-to-RLOC mapping resolutions. <ul style="list-style-type: none">The locator address of the map resolver may be an IPv4 or IPv6 address. Note You can configure up to 8 map resolvers if multiple map resolvers are available.
Step 15	ipv4 itr Example: Device(config-router-lisp)# ipv4 itr	Enables LISP ingress tunnel router (ITR) functionality for the IPv4 address family.
Step 16	ipv4 etr map-server map-server-address key authentication-key Example: Device(config-router-lisp)# ipv4 etr map-server 172.20.5.5 key mskey	Configures the IPv4 or IPv6 locator address of the LISP map server to be used by the egress tunnel router (ETR) when registering IPv4 endpoint identifiers (EIDs).
Step 17	ipv4 etr Example: Device(config-router-lisp)# ipv4 etr	Enables LISP ETR functionality for the IPv4 address family.
Step 18	exit Example: Device(config-router-lisp)# exit	Exits LISP configuration mode and returns to global configuration mode.
Step 19	interface type number Example: Device(config)# interface FastEthernet 1/4	Specifies the interface type and number and enters interface configuration mode.
Step 20	ip address ip-address mask Example: Device(config-if)# ip address 192.0.2.21 255.255.255.0	Configures an IPv4 address for the interface.
Step 21	lisp mobility dynamic-eid-name Example: Device(config-if)# lisp mobility VMs	Allows EID mobility on the interface and specifies the name of the dynamic EID.

	Command or Action	Purpose
Step 22	lisp extended-subnet-mode Example: Device(config-if)# lisp extended-subnet-mode	Enables extended subnet mode on the interface.
Step 23	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring xTR

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router lisp**
4. **locator-set** *locator-set-name*
5. *ipv4-address* **priority** *priority-locator* **weight** *locator-weight*
6. Repeat Step 5 to configure another locator entry.
7. **exit**
8. **eid-table default instance-id** *id*
9. **database-mapping** *dynamic-eid-prefix/prefix-length* **locator-set** *name*
10. **exit**
11. **ipv4 itr map-resolver** *map-resolver-address*
12. **ipv4 itr**
13. **ipv4 etr map-server** *map-server-address* **key** *authentication-key*
14. **ipv4 etr**
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router lisp Example: Device(config)# router lisp	Enters LISP configuration mode.

	Command or Action	Purpose
Step 4	locator-set <i>locator-set-name</i> Example: Device(config-router-lisp)# locator-set Site3RLOCS	Specifies a locator set and enters LISP locator-set configuration mode.
Step 5	<i>ipv4-address</i> priority <i>priority-locator</i> weight <i>locator-weight</i> Example: Device(config-router-lisp-locator-set)# 203.0.113.2 priority 10 weight 50	Configures the LISP locator set. The LISP locator set is the set of addresses used by the gateway xTR while encapsulating/decapsulating LISP traffic from and to the endpoint identifier (EID).
Step 6	Repeat Step 5 to configure another locator entry.	—
Step 7	exit Example: Device(config-router-lisp-locator-set)# exit	Exits LISP locator set configuration mode and returns to LISP configuration mode.
Step 8	eid-table default instance-id <i>id</i> Example: Device(config-router-lisp)# eid-table default instance-id 0	Configures an association between the default VRF table and a LISP instance ID, and enters EID table configuration mode.
Step 9	database-mapping <i>dynamic-eid-prefix/prefix-length</i> locator-set <i>name</i> Example: Device(config-router-lisp-eid-table)# database-mapping 198.51.100.0/24 locator-set Site3RLOCS	Configures an IPv4 mapping relationship and an associated traffic policy for the LISP Virtual Machine (VM)-mobility (dynamic EID) policy. Note You can enter the limit dynamic value keyword to limit the number of discoverable dynamic EIDs. However, if you have enabled debug mode (using the service internal command), then the number of discoverable dynamic EIDs will be increased to a fixed value of 65535.
Step 10	exit Example: Device(config-router-lisp-eid-table)# exit	Exits EID table configuration mode and returns to LISP configuration mode.
Step 11	ipv4 itr map-resolver <i>map-resolver-address</i> Example: Device(config-router-lisp)# ipv4 itr map-resolver 172.20.5.5	Configures a locator address for the LISP map resolver to which this router will send map request messages for IPv4 EID-to-RLOC mapping resolutions. <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. Note You can configure up to 8 map resolvers if multiple map resolvers are available.

	Command or Action	Purpose
Step 12	ipv4 itr Example: Device(config-router-lisp)# ipv4 itr	Enables LISP ITR functionality for an IPv4 address family.
Step 13	ipv4 etr map-server map-server-address key authentication-key Example: Device(config-router-lisp)# ipv4 etr map-server 172.20.5.5 key k3	Configures IPv4 locator address of the LISP map server to be used by the egress tunnel router (ETR) when registering for IPv4 endpoint identifiers (EIDs).
Step 14	ipv4 etr Example: Device(config-router-lisp)# ipv4 etr	Enables LISP ETR functionality for an IPv4 address family.
Step 15	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring Map Server Map Resolver

SUMMARY STEPS

1. enable
2. configure terminal
3. router lisp
4. site *site-name*
5. authentication-key *password*
6. eid-prefix *eid-prefix* accept-more-specifics
7. exit
8. Repeat Step 4 to Step 7 to configure another LISP site.
9. ipv4 map-server
10. ipv4 map-resolver
11. end

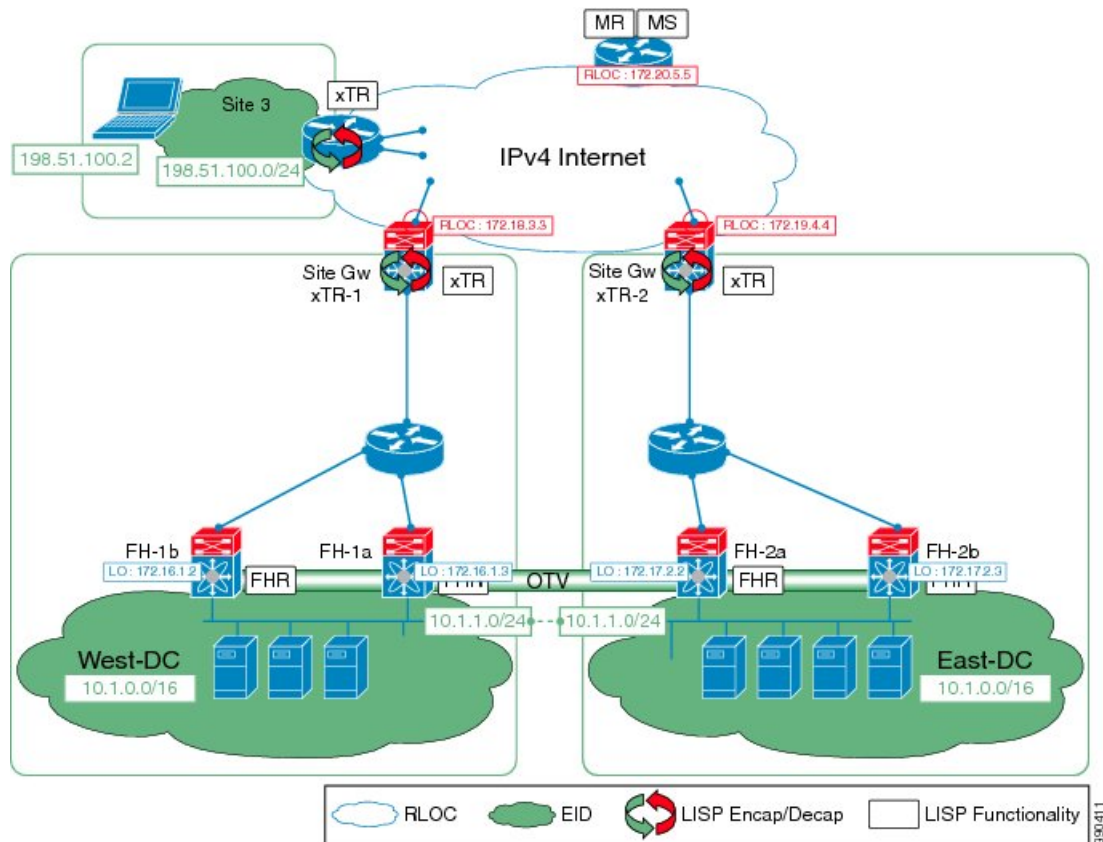
DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# configure terminal	
Step 3	router lisp Example: Device(config)# router lisp	Enters Locator ID/Separation Protocol (LISP) configuration mode.
Step 4	site site-name Example: Device(config-router-lisp)# site EastWestDC	Configures a LISP site and enters LISP site configuration mode on a LISP map server.
Step 5	authentication-key password Example: Device(config-router-lisp-site)# authentication-key k	Configures the password used to create the Hash-based Message Authentication Code (HMAC) Secure Hash Algorithm (SHA-1) hash for authenticating the map-register message sent by an egress tunnel router (ETR) when registering with the map server.
Step 6	eid-prefix eid-prefix accept-more-specifics Example: Device(config-router-lisp-site)# eid-prefix 10.1.0.0/16 accept-more-specifics	Configures a list of endpoint identifier (EID) prefixes that are allowed in a map-register message sent by an ETR when registering with the map server. Specifies that any EID prefix that is more specific than the EID prefix configured is accepted and tracked.
Step 7	exit Example: Device(config-router-lisp-site)# exit	Exits LISP site configuration mode and returns to LISP configuration mode.
Step 8	Repeat Step 4 to Step 7 to configure another LISP site.	—
Step 9	ipv4 map-server Example: Device(config-router-lisp)# ipv4 map-server	Configures a device to act as an IPv4 LISP map server.
Step 10	ipv4 map-resolver Example: Device(config-router-lisp)# ipv4 map-resolver	Configures a device to act as an IPv4 LISP map resolver.
Step 11	end Example: Device(config-router-lisp)# end	Exits LISP configuration mode and returns to privileged EXEC mode.

Configuration Examples for LISP ESM Multihop Mobility

Figure 44: LISP ESM Multihop Topology



The examples below show the complete configuration for the LISP topology illustrated in the figure above.

Example: First-Hop Router Configuration

```

Device# configure terminal
Device(config)# router lisp
Device(config-router-lisp)# locator-set WestDC
Device(config-router-lisp-locator-set)# 172.16.1.2 priority 10 weight 50
Device(config-router-lisp-locator-set)# 172.17.2.3 priority 10 weight 50
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# eid-table default instance-id 0
Device(config-router-lisp-eid-table)# dynamic-eid VMs
Device(config-router-lisp-eid-table-dynamic-eid)# database-mapping 10.1.1.0/24 locator-set
WestDC
Device(config-router-lisp-eid-table-dynamic-eid)# eid-notify 192.0.2.21 key k
Device(config-router-lisp-eid-table-dynamic-eid)# map-notify-group 224.0.0.0
Device(config-router-lisp-eid-table-dynamic-eid)# exit
Device(config-router-lisp-eid-table)# exit
Device(config-router-lisp)# exit
Device(config)# interface Vlan11
Device(config-if)# lisp mobility VMs
  
```

```
Device(config-if)# lisp extended-subnet-mode
Device(config-if)# ip address 10.1.1.2 255.255.255.0
Device(config-if)# standby 1 ip 10.1.1.1
```

Example: Site Gateway xTR Configuration

```
Device> enable
Device# configure terminal
Device (config)# router lisp
Device(config-router-lisp)# locator-set WestDC
Device(config-router-lisp-locator-set) # 172.18.3.3 priority 10 weight 50
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# eid-table default instance-id 0
Device(config-router-lisp-eid-table)# database-mapping 10.1.0.0/16 locator-set WestDC
Device(config-router-lisp-eid-table)# dynamic-eid VMs
Device(config-router-lisp-eid-table-dynamic-eid)# database-mapping 10.1.1.0/24 locator-set
WestDC
Device(config-router-lisp-eid-table-dynamic-eid)# eid-notify authentication-key k
Device(config-router-lisp-eid-table-dynamic-eid)# exit
Device(config-router-lisp-eid-table)# exit
Device(config-router-lisp)# ipv4 itr map-resolver 172.20.5.5
Device(config-router-lisp)# ipv4 itr
Device(config-router-lisp)# ipv4 etr map-server 172.20.5.5 key k
Device(config-router-lisp)# ipv4 etr
Device(config-router-lisp)# exit
Device(config)# interface FastEthernet1/4
Device(config-if)# ip address 192.0.2.21 255.255.255.0
Device(config-if)# lisp mobility VMs
Device(config-if)# lisp extended-subnet-mode
```

Example: xTR Configuration

```
Device> enable
Device# configure terminal
Device (config)# router lisp
Device(config-router-lisp)# locator-set Site3RLOCS
Device(config-router-lisp-locator-set) # 203.0.113.2 priority 10 weight 50
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# eid-table default instance-id 0
Device(config-router-lisp-eid-table)# database-mapping 198.51.100.0/24 locator-set Site3RLOCS
Device(config-router-lisp-eid-table)# exit
Device(config-router-lisp)# ipv4 itr map-resolver 172.20.5.5
Device(config-router-lisp)# ipv4 itr
Device(config-router-lisp)# ipv4 etr map-server 172.20.5.5 key k3
Device(config-router-lisp)# ipv4 etr
```

Example: Map Server Map Resolver Configuration

```
Device> enable
Device# configure terminal
Device (config)# router lisp
Device(config-router-lisp)# site EastWestDC
Device(config-router-lisp-site)# authentication-key k
Device(config-router-lisp-site)# eid-prefix 10.1.0.0/16 accept-more-specifics
Device(config-router-lisp-site)# exit
```

```
Device(config-router-lisp)# ipv4 map-server
Device(config-router-lisp)# ipv4 map-resolver
```

Additional References for LISP ESM Multihop Mobility

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Locator/ID Separation Protocol (LISP) commands	Cisco IOS IP Routing: LISP Command Reference

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP ESM Multihop Mobility

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Feature Name	Release	Feature Information
LISP ESM Multihop Mobility		The LISP ESM Multihop Mobility feature separates the Locator/ID Separation Protocol (LISP) dynamic host detection function from the LISP encapsulation/decapsulation function within a LISP topology.



CHAPTER 10

LISP Support for Disjoint RLOC Domains

The Locator/ID Separation Protocol (LISP) implements a “level of indirection” that enables a new IP routing architecture. LISP separates IP addresses into two namespaces: Endpoint Identifiers (EIDs), which are assigned to end-hosts, and Routing Locators (RLOCs), which are assigned to devices that make up the global routing system.

The LISP Support for Disjoint RLOC Domains feature enables LISP-to-LISP communication between LISP sites that are connected to different RLOC spaces but have no connectivity to each other. One example of disjointed RLOC space is that of between the IPv4 Internet and IPv6 Internet. When one LISP site has IPv4-only RLOC connectivity and the second site has IPv6-only RLOC connectivity, these sites can still communicate via LISP using the LISP Support for Disjoint RLOC Domains feature.

- [Finding Feature Information, on page 191](#)
- [Prerequisites for LISP Support for Disjoint RLOC Domains, on page 191](#)
- [Restrictions for LISP Support for Disjoint RLOC Domains, on page 192](#)
- [Information About LISP Support for Disjoint RLOC Domains, on page 192](#)
- [How to configure LISP Support for Disjoint RLOC Domains, on page 194](#)
- [Verifying LISP Support for Disjoint RLOC Domains, on page 205](#)
- [Configuration Examples for LISP Support for Disjoint RLOC Domains, on page 206](#)
- [Additional References for LISP Support for Disjoint RLOC Domains, on page 210](#)
- [Feature Information for LISP Support for Disjoint RLOC Domains, on page 211](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for LISP Support for Disjoint RLOC Domains

Map servers and re-encapsulating tunnel routers (RTRs) must have connectivity to all locator spaces that are being joined.

Restrictions for LISP Support for Disjoint RLOC Domains

Map servers and re-encapsulating tunnel routers (RTRs) cannot join more than eight locator scopes.

Information About LISP Support for Disjoint RLOC Domains

LISP Support for Disjoint RLOC Domains Overview

The fundamental principal of any network is that routing and reachability must exist between all devices that make up the total network system. There are many network systems, public and private, for which internetwork connectivity is not directly available. A few examples include:

- IPv4 Internet and IPv6 Internet.
- An IPv4 Multiprotocol Label Switching (MPLS) VPN from service provider A and an IPv4 MPLS VPN from service provider B.
- An IPv4 MPLS VPN from service provider A and IPv4 Internet.

When some sites within a network connect to one routing domain and other sites connect to another routing domain, a gateway function must be provided to facilitate connectivity between these disjointed routing domains. In traditional routing architectures, providing connectivity between disjointed routing domains can be quite complex.

The inherent property of Locator/ID Separation Protocol (LISP), which separates IP addresses into two namespaces, endpoint identifiers (EIDs) and routing locators (RLOCs), also gives it the ability to connect disjointed RLOC domains. The LISP Support for Disjoint RLOC Domains feature provides simplified configuration mechanisms that enable this capability. The key components are new control plane configuration options on the LISP map server, and a functionality called re-encapsulating tunnel router (RTR), which provides data plane connectivity between disjointed locator spaces.

LISP Map Server

The key concept in the LISP Support for Disjoint RLOC Domains feature is the recognition that the LISP Mapping System has full knowledge of all LISP sites. When a LISP site registers with a map server, the registration message not only provides information about the EID space that the site is authoritative for, but it also provides information about its own RLOCs.

The LISP Support for Disjoint RLOC Domains feature provides new configuration options to define within the map server the routing locator scopes that LISP sites can connect to. Once defined, the map server automatically determines whether individual sites have common or disjoint locator connectivity between themselves. The map server then uses this knowledge when handling Map-Request messages to determine how to inform LISP sites to communicate with each other. Map-Request messages contain both source and destination EID information. When a map server receives a Map-Request message, it compares the RLOCs associated with the source EID and destination EID contained with the Map-Request message against the configured locator scopes.

- If the ingress tunnel router (ITR) (source EID) and egress tunnel router (ETR) (destination EID) share at least one RLOC in a common locator scope, the map server forwards the Map-Request message to the ETR as normal. In this case, the ETR is capable of generating a Map-Reply message that is sent back to the ITR since it has reachability across (at least one) common locator space.

How to configure LISP Support for Disjoint RLOC Domains

Configuring xTR

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **ipv6 address** *ipv6-address/ipv6-prefix*
6. **interface** *type number*
7. **ip address** *ip-address mask*
8. **router lisp**
9. **locator-set** *locator-set-name*
10. *ipv4-address* **priority** *priority-locator* **weight** *locator-weight*
11. *ipv6-address* **priority** *priority-locator* **weight** *locator-weight*
12. **exit**
13. **eid-table default instance-id** *id*
14. **database-mapping** *dynamic-eid-prefix/prefix-length* **locator-set** *name*
15. **database-mapping** *dynamic-eid-prefix/prefix-length* **locator-set** *name*
16. **exit**
17. **ipv4 itr map-resolver** *map-resolver-address*
18. **ipv4 itr**
19. **ipv4 etr map-server** *map-server-address* **key** *authentication-key*
20. **ipv4 etr**
21. **ipv6 itr map-resolver** *map-resolver-address*
22. **ipv6 itr**
23. **ipv6 etr map-server** *map-server-address* **key** *authentication-key*
24. **ipv6 etr**
25. **exit**
26. **ip route** *prefix mask ip-address*
27. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# configure terminal	
Step 3	interface <i>type number</i> Example: Device(config)# interface loopback0	Specifies the interface type and number and enters interface configuration mode.
Step 4	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.10.10.4 255.255.255.0	Configures an IPv4 address for the interface.
Step 5	ipv6 address <i>ipv6-address/ipv6-prefix</i> Example: Device(config-if)# ipv6 address 2001:DB8:0:ABCD::1/64	Configures an IPv6 address for the interface.
Step 6	interface <i>type number</i> Example: Device(config)# interface ethernet0/0	Specifies the interface type and number and enters interface configuration mode.
Step 7	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.0.4.1 255.255.255.252	Configures an IPv4 address for the interface.
Step 8	router lisp Example: Device(config-if)# router lisp	Enters LISP configuration mode.
Step 9	locator-set <i>locator-set-name</i> Example: Device(config-router-lisp)# locator-set R4	Specifies a locator set and enters LISP locator set configuration mode.
Step 10	ipv4-address priority <i>priority-locator weight</i> locator-weight Example: Device(config-router-lisp-locator-set)# 10.0.4.1 priority 1 weight 1	Configures the LISP locator set. The LISP locator set is the set of addresses the first-hop router uses when communicating with the gateway xTR. You can configure each IPv4 locator address by creating a locator entry with assigned priority and weight.
Step 11	ipv6-address priority <i>priority-locator weight</i> locator-weight Example: Device(config-router-lisp-locator-set)# 2001:DB8:4::2 priority 1 weight 1	Configures the LISP locator set. The LISP locator set is the set of addresses the first-hop router uses when communicating with the gateway xTR. You can configure each IPv6 locator address by creating a locator entry with assigned priority and weight.
Step 12	exit Example:	Exits LISP locator set configuration mode and returns to LISP configuration mode.

	Command or Action	Purpose
	<code>Device(config-router-lisp-locator-set)# exit</code>	
Step 13	eid-table default instance-id <i>id</i> Example: <code>Device(config-router-lisp)# eid-table default instance-id 0</code>	Configures an association between the default (global) routing table and a LISP instance ID, and enters EID table configuration mode.
Step 14	database-mapping <i>dynamic-eid-prefix/prefix-length</i> locator-set <i>name</i> Example: <code>Device(config-router-lisp-eid-table)# database-mapping 10.10.10.0/24 locator-set R4</code>	Configures an IPv4/IPv6 mapping relationship and an associated traffic policy (as defined in the locator set) for this LISP site.
Step 15	database-mapping <i>dynamic-eid-prefix/prefix-length</i> locator-set <i>name</i> Example: <code>Device(config-router-lisp-eid-table)# database-mapping 2001:DB8::/48 locator-set R4</code>	Configures an IPv4/IPv6 mapping relationship and an associated traffic policy (as defined in the locator set) for this LISP site.
Step 16	exit Example: <code>Device(config-router-lisp-eid-table)# exit</code>	Exits EID table configuration mode and returns to LISP configuration mode.
Step 17	ipv4 itr map-resolver <i>map-resolver-address</i> Example: <code>Device(config-router-lisp)# ipv4 itr map-resolver 10.0.2.1</code>	Configures a locator address for the LISP map resolver to which this device will send Map-Request messages for IPv4 endpoint identifier-to-routing locator (EID-to-RLOC) mapping resolutions. <ul style="list-style-type: none"> • The locator address of the map resolver may be an IPv4 or IPv6 address. <p>Note You can configure up to eight map resolvers if multiple map resolvers are available.</p>
Step 18	ipv4 itr Example: <code>Device(config-router-lisp)# ipv4 itr</code>	Enables LISP ingress tunnel router (ITR) functionality for an IPv4 address family.
Step 19	ipv4 etr map-server <i>map-server-address</i> key <i>authentication-key</i> Example: <code>Device(config-router-lisp)# ipv4 etr map-server 10.0.2.1 key R4KEY</code>	Configures the IPv4 locator address of the LISP map server to be used by the egress tunnel router (ETR) when registering itself for IPv4 endpoint identifiers (EIDs).
Step 20	ipv4 etr Example: <code>Device(config-router-lisp)# ipv4 etr</code>	Enables LISP ETR functionality for an IPv4 address family.

	Command or Action	Purpose
Step 21	ipv6 itr map-resolver <i>map-resolver-address</i> Example: Device(config-router-lisp)# ipv6 itr map-resolver 10.0.2.1	Configures a locator address for the LISP map resolver to which this router will send Map-Request messages for IPv6 EID-to-RLOC mapping resolutions. <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. Note You can configure up to eight map resolvers if multiple map resolvers are available.
Step 22	ipv6 itr Example: Device(config-router-lisp)# ipv6 itr	Enables LISP ITR functionality for an IPv6 address family.
Step 23	ipv6 etr map-server <i>map-server-address key authentication-key</i> Example: Device(config-router-lisp)# ipv6 etr map-server 10.0.2.1 key R4KEY	Configures the IPv6 locator address for the LISP map server to be used by the ETR when registering for IPv6 EIDs.
Step 24	ipv6 etr Example: Device(config-router-lisp)# ipv6 etr	Enables LISP ETR functionality for an IPv6 address family.
Step 25	exit Example: Device(config-router-lisp)# exit	Exits LISP configuration mode and returns to global configuration mode.
Step 26	ip route <i>prefix mask ip-address</i> Example: Device(config)# ip route 0.0.0.0 0.0.0.0 10.0.4.2	Establishes static routes to the next hop destination.
Step 27	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring MSMR

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **ipv6 address** *ipv6-address/ipv6-prefix*

6. **router lisp**
7. **locator-set** *locator-set-name*
8. *ipv4-address* **priority** *priority-locator* **weight** *locator-weight*
9. **exit**
10. Repeat Step 7 to Step 9 to specify and configure another locator set.
11. **locator-scope** *name*
12. **rtr-locator-set** *locator-set-name*
13. **rloc-prefix** *ipv4-rloc-prefix*
14. **exit**
15. Repeat Step 11 to Step 14 to specify and configure another locator scope.
16. **site** *site-name*
17. **authentication-key** *password*
18. **eid-prefix** *ipv4-eid-prefix*
19. **eid-prefix** *ipv6-eid-prefix*
20. **exit**
21. Repeat Step 16 to Step 20 to configure another LISP site on the map server.
22. **ipv4 map-server**
23. **ipv6 map-server**
24. **ipv4 map-resolver**
25. **ipv6 map-resolver**
26. **exit**
27. **ip route** *prefix mask ip-address*
28. **ipv6 route** *ipv6-prefix/prefix-length ipv6-address*
29. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Device(config)# interface ethernet0/0	Specifies the interface type and number and enters interface configuration mode.
Step 4	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.0.2.1 255.255.255.252	Configures an IPv4 address for the interface.

	Command or Action	Purpose
Step 5	ipv6 address <i>ipv6-address/ipv6-prefix</i> Example: Device(config-if)# ipv6 address 2001:DB8:1::1/64	Configures an IPv6 address for the interface.
Step 6	router lisp Example: Device(config-if)# router lisp	Enters LISP configuration mode.
Step 7	locator-set <i>locator-set-name</i> Example: Device(config-router-lisp)# locator-set rtr-set1	Specifies a locator set and enters LISP locator set configuration mode.
Step 8	ipv4-address priority priority-locator weight <i>locator-weight</i> Example: Device(config-router-lisp-locator-set)# 10.0.3.1 priority 1 weight 1	Configures the LISP locator set. The LISP locator set is the set of addresses the first-hop router uses when communicating with the gateway xTR. You can configure each locator address by creating a locator entry with assigned priority and weight.
Step 9	exit Example: Device(config-router-lisp-locator-set)# exit	Exits LISP locator set configuration mode and returns to LISP configuration mode.
Step 10	Repeat Step 7 to Step 9 to specify and configure another locator set.	—
Step 11	locator-scope <i>name</i> Example: Device(config-router-lisp)# locator-scope s1	Specifies the locator scope and enters locator scope configuration mode.
Step 12	rtr-locator-set <i>locator-set-name</i> Example: Device(config-router-lisp-locator-scope)# rtr-locator-set rtr-set1	Specifies the locator set of re-encapsulating tunnel router (RTR) to use in proxy reply for disjoint/cross address family routing locator (RLOC).
Step 13	rloc-prefix <i>ipv4-rloc-prefix</i> Example: Device(config-router-lisp-locator-scope)# rloc-prefix 0.0.0.0/0	Specifies the RLOC prefix to check against ingress tunnel router (ITR) RLOC and egress tunnel router (ETR) RLOC.
Step 14	exit Example: Device(config-router-lisp-locator-set)# exit	Exits LISP locator set configuration mode and returns to LISP configuration mode.
Step 15	Repeat Step 11 to Step 14 to specify and configure another locator scope.	—

	Command or Action	Purpose
Step 16	site <i>site-name</i> Example: Device(config-router-lisp)# site R4	Configures a LISP site on a map server and enters LISP site configuration mode.
Step 17	authentication-key <i>password</i> Example: Device(config-router-lisp-site)# authentication-key R4KEY	Specifies the authentication key that the LISP site uses.
Step 18	eid-prefix <i>ipv4-eid-prefix</i> Example: Device(config-router-lisp-site)# eid-prefix 10.10.10.0/24	Specifies a site IPv4 EID prefix.
Step 19	eid-prefix <i>ipv6-eid-prefix</i> Example: Device(config-router-lisp-site)# eid-prefix 2001:DB8::/48	Specifies a site IPv6 EID address prefix.
Step 20	exit Example: Device(config-router-lisp-site)# exit	Exits LISP site configuration mode and returns to LISP configuration mode.
Step 21	Repeat Step 16 to Step 20 to configure another LISP site on the map server.	—
Step 22	ipv4 map-server Example: Device(config-router-lisp)# ipv4 map-server	Enables IPv4 map server functionality.
Step 23	ipv6 map-server Example: Device(config-router-lisp)# ipv6 map-server	Enables IPv6 map server functionality.
Step 24	ipv4 map-resolver Example: Device(config-router-lisp)# ipv4 map-resolver	Enables IPv4 map resolver functionality.
Step 25	ipv6 map-resolver Example: Device(config-router-lisp)# ipv6 map-resolver	Enables IPv6 map resolver functionality.
Step 26	exit Example: Device(config-router-lisp)# exit	Exits LISP configuration mode and returns to global configuration mode.

	Command or Action	Purpose
Step 27	ip route <i>prefix mask ip-address</i> Example: Device(config)# ip route 0.0.0.0 0.0.0.0 10.0.2.2	Establishes static routes to the next hop destination.
Step 28	ipv6 route <i>ipv6-prefix/prefix-length ipv6-address</i> Example: Device(config)# ipv6 route ::/0 2001:DB8:1::ABCD	Establishes static routes to the next hop destination.
Step 29	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring RTR

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **ipv6 address** *ipv6-address/ipv6-prefix*
6. **router lisp**
7. **locator-set** *locator-set-name*
8. *ipv4-address* **priority** *priority-locator* **weight** *locator-weight*
9. *ipv6-address* **priority** *priority-locator* **weight** *locator-weight*
10. **exit**
11. **map-request itr-rlocs** *locator-set-name*
12. **eid-table default instance-id** *id*
13. **map-cache** *ipv4-EID-prefix* **map-request**
14. **map-cache** *ipv6-EID-prefix* **map-request**
15. **exit**
16. **ipv4 map-request-source** *source-address*
17. **ipv4 map-cache-limit** *cache-limit*
18. **ipv4 proxy-etr**
19. **ipv4 proxy-itr** *ipv4-local-locator ipv6-local-locator*
20. **ipv4 itr map-resolver** *map-resolver-address*
21. **ipv6 map-request-source** *source-address*
22. **ipv6 map-cache-limit** *cache-limit*
23. **ipv6 proxy-etr** *cache-limit*
24. **ipv6 proxy-itr** *ipv6-local-locator ipv4-local-locator*
25. **ipv6 itr map-resolver** *map-resolver-address*
26. **exit**

27. **ip route** *prefix mask ip-address*
28. **ipv6 route** *ipv6-prefix/prefix-length ipv6-address*
29. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Device(config)# interface ethernet0/0	Specifies the interface type and number and enters interface configuration mode.
Step 4	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.0.3.1 255.255.255.252	Configures an IPv4 address for the interface.
Step 5	ipv6 address <i>ipv6-address/ipv6-prefix</i> Example: Device(config-if)# ipv6 address 2001:DB8:2::1/64	Configures an IPv6 address for the interface.
Step 6	router lisp Example: Device(config-if)# router lisp	Enters LISP configuration mode.
Step 7	locator-set <i>locator-set-name</i> Example: Device(config-router-lisp)# locator-set setALL	Specifies a locator set and enters LISP locator set configuration mode.
Step 8	ipv4-address priority <i>priority-locator</i> weight <i>locator-weight</i> Example: Device(config-router-lisp-locator-set)# 10.0.3.1 priority 1 weight 1	Configures an IPv4 or IPv6 address and policy for the re-encapsulation tunnel router (RTR).
Step 9	ipv6-address priority <i>priority-locator</i> weight <i>locator-weight</i> Example: Device(config-router-lisp-locator-set)# 2001:DB8:2::1 priority 1 weight 1	Configures an IPv4 or IPv6 address and policy for the RTR.

	Command or Action	Purpose
Step 10	exit Example: Device(config-router-lisp-locator-set)# exit	Exits LISP locator set configuration mode and returns to LISP configuration mode.
Step 11	map-request itr-rlocs locator-set-name Example: Device(config-router-lisp)# map-request itr-rlocs setALL	Configures the locator set to be used as routing locators (RLOCs) in the ingress tunnel router (ITR) RLOC field of Map-Request messages sent from the RTR.
Step 12	eid-table default instance-id id Example: Device(config-router-lisp)# eid-table default instance-id 0	Configures an association between the default (global) routing table and a LISP instance ID, and enters EID table configuration mode.
Step 13	map-cache ipv4-EID-prefix map-request Example: Device(config-router-lisp-eid-table)# map-cache 0.0.0.0/0 map-request	Configures static endpoint identifier-to-routing locator (EID-to-RLOC) mappings for an ITR and enables sending of Map-Request message for a LISP destination EID.
Step 14	map-cache ipv6-EID-prefix map-request Example: Device(config-router-lisp-eid-table)# map-cache ::/0 map-request	Configures static EID-to-RLOC mappings for an ITR and enables sending of Map-Request message for a LISP destination EID.
Step 15	exit Example: Device(config-router-lisp-eid-table)# exit	Exits LISP EID table configuration mode and returns to LISP configuration mode.
Step 16	ipv4 map-request-source source-address Example: Device(config-router-lisp)# ipv4 map-request-source 10.0.3.1	Specifies the IPv4 source address to be used in LISP IPv4 Map-Request messages. The ITR RLOCs configured under Steps 7 through 10, and Step 11 take precedence. However, this step (16) is still required.
Step 17	ipv4 map-cache-limit cache-limit Example: Device(config-router-lisp)# ipv4 map-cache-limit 100000	(Optional) Specifies maximum number of IPv4 LISP map cache entries allowed to be stored on the router. The valid range is from 0 to 100000.
Step 18	ipv4 proxy-etr Example: Device(config-router-lisp)# ipv4 proxy-etr	Configures a device to act as an IPv4 LISP proxy egress tunnel router (PETR).
Step 19	ipv4 proxy-itr ipv4-local-locator ipv6-local-locator Example: Device(config-router-lisp)# ipv4 proxy-itr 10.0.3.1 2001:DB8:2::1	Configures this device to act as an IPv4 proxy ingress tunnel router (PITR), and configures the IPv4 and IPv6 locator addresses used as a source address for encapsulation of data packets.

	Command or Action	Purpose
Step 20	ipv4 itr map-resolver map-resolver-address Example: Device(config-router-lisp)# ipv4 itr map-resolver 10.0.2.1 Device(config-router-lisp)# ipv4 itr map-resolver 2001:DB8:1::1	Configures a locator address for the LISP map resolver to which this device will send Map-Request messages for IPv4 EID-to-RLOC mapping resolutions. <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. Note You can configure up to 8 map resolvers if multiple map resolvers are available.
Step 21	ipv6 map-request-source source-address Example: Device(config-router-lisp)# ipv6 map-request-source 2001:DB8:2::1	The ITR RLOCs configured under Steps 7 through 10, and Step 11 take precedence. However, this step (16) is still required.
Step 22	ipv6 map-cache-limit cache-limit Example: Device(config-router-lisp)# ipv6 map-cache-limit 100000	(Optional) Specifies the maximum number of IPv6 LISP map cache entries allowed to be stored on the device. The valid range is from 0 to 100000.
Step 23	ipv6 proxy-etr cache-limit Example: Device(config-router-lisp)# ipv6 proxy-etr	Configures a device to act as an IPv6 LISP PETR.
Step 24	ipv6 proxy-itr ipv6-local-locator ipv4-local-locator Example: Device(config-router-lisp)# ipv6 proxy-itr 2001:DB8:2::1 10.0.3.1	Configures this device to act as an IPv6 PITR, and configures the IPv4 and IPv6 locator addresses used as a source address for encapsulation of data packets.
Step 25	ipv6 itr map-resolver map-resolver-address Example: Device(config-router-lisp)# ipv6 itr map-resolver 10.0.2.1 Device(config-router-lisp)# ipv6 itr map-resolver 2001:DB8:1::1	Configures a locator address for the LISP map resolver to which this router will send Map-Request messages for IPv6 EID-to-RLOC mapping resolutions. <ul style="list-style-type: none"> The locator address of the map resolver may be an IPv4 or IPv6 address. Note You can configure up to eight map resolvers if multiple map resolvers are available.
Step 26	exit Example: Device(config-router-lisp)# exit	Exits LISP configuration mode and returns to global configuration mode.
Step 27	ip route prefix mask ip-address Example: Device(config)# ip route 0.0.0.0 0.0.0.0 10.0.3.2	Establishes static routes to the next hop destination.

	Command or Action	Purpose
Step 28	ipv6 route <i>ipv6-prefix/prefix-length ipv6-address</i> Example: Device(config)# ipv6 route ::/0 2001:DB8:ABCD::1	Establishes static routes to the next hop destination.
Step 29	end Example: Device(config)# end	Returns to privileged EXEC mode.

Verifying LISP Support for Disjoint RLOC Domains

SUMMARY STEPS

1. enable
2. show ip lisp database
3. show ipv6 lisp database
4. show lisp site detail
5. show ip lisp map-cache
6. show ipv6 lisp map-cache

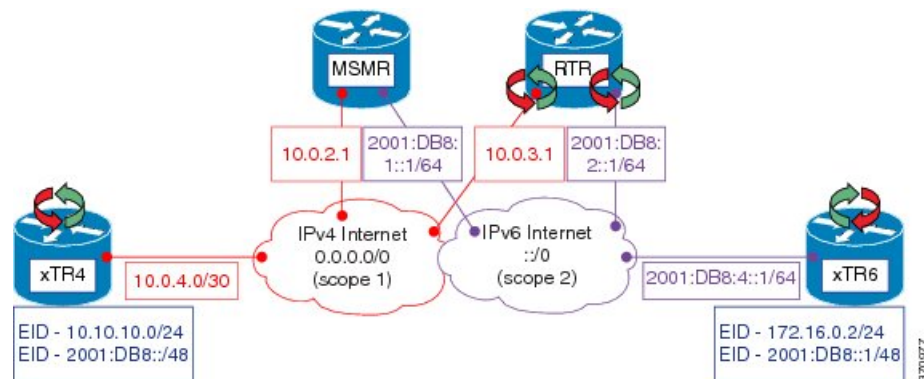
DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip lisp database Example: Device# show ip lisp database	Displays Locator/ID Separation Protocol (LISP) egress tunnel router (ETR) configured local IPv4 endpoint identifier (EID) prefixes and associated locator sets.
Step 3	show ipv6 lisp database Example: Device# show ipv6 lisp database	Displays LISP ETR configured local IPv6 EID prefixes and associated locator sets.
Step 4	show lisp site detail Example: Device# show lisp site detail	Displays details of LISP sites configured on a LISP map server.
Step 5	show ip lisp map-cache Example: Device# show ip lisp map-cache	Displays the current dynamic and static IPv4 endpoint identifier-to-routing locator (EID-to-RLOC) map cache entries.

	Command or Action	Purpose
Step 6	show ipv6 lisp map-cache Example: Device# show ipv6 lisp map-cache	Displays the current dynamic and static IPv6 EID-to-RLOC map cache entries.

Configuration Examples for LISP Support for Disjoint RLOC Domains

Figure 46: LISP - Disjoint RLOC Domains topology



The examples below show the complete configuration for the LISP topology illustrated in the figure above.

Example: Configuring xTR

The following example shows how to configure xTR4:

```

Device> enable
Device# configure terminal
Device(config)# interface loopback0
Device(config-if)# ip address 10.10.10.4 255.255.255.0
Device(config-if)# ipv6 address 2001:DB8:0:ABCD::1/64
Device(config-if)# interface ethernet0/0
Device(config-if)# ip address 10.0.4.1 255.255.255.252
Device(config-if)# router lisp
Device(config-router-lisp)# locator-set R4
Device(config-router-lisp-locator-set)# 10.0.4.1 priority 1 weight 1
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# eid-table default instance-id 0
Device(config-router-lisp-eid-table)# database-mapping 10.10.10.0/24 locator-set R4
Device(config-router-lisp-eid-table)# database-mapping 2001:DB8::/48 locator-set R4
Device(config-router-lisp-eid-table)# exit
Device(config-router-lisp)# ipv4 itr map-resolver 10.0.2.1
Device(config-router-lisp)# ipv4 itr
Device(config-router-lisp)# ipv4 etr map-server 10.0.2.1 key R4KEY
Device(config-router-lisp)# ipv4 etr
Device(config-router-lisp)# ipv6 itr map-resolver 10.0.2.1
Device(config-router-lisp)# ipv6 itr
  
```



```

Device(config-router-lisp)# ipv6 etr map-server 10.0.2.1 key R4KEY
Device(config-router-lisp)# ipv6 etr
Device(config-router-lisp)# exit
Device(config)# ip route 0.0.0.0 0.0.0.0 10.0.4.2

```

The following example shows how to configure xTR6:

```

Device> enable
Device# configure terminal
Device(config)# interface loopback0
Device(config-if)# ip address 172.16.0.4 255.255.255.0
Device(config-if)# ipv6 address 2001:DB8::4/64
Device(config-if)# interface ethernet0/0
Device(config-if)# ipv6 address 2001:DB8:4::2/64
Device(config-if)# router lisp
Device(config-router-lisp)# locator-set R6
Device(config-router-lisp-locator-set)# 2001:DB8:4::2 priority 1 weight 1
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# eid-table default instance-id 0
Device(config-router-lisp-eid-table)# database-mapping 172.16.0.2/24 locator-set R4
Device(config-router-lisp-eid-table)# database-mapping 2001:DB8::1/48 locator-set R4
Device(config-router-lisp-eid-table)# exit
Device(config-router-lisp)# ipv4 itr map-resolver 2001:DB8:3::2
Device(config-router-lisp)# ipv4 itr
Device(config-router-lisp)# ipv4 etr map-server 2001:DB8:3::2 key R4KEY
Device(config-router-lisp)# ipv4 etr
Device(config-router-lisp)# ipv6 itr map-resolver 2001:DB8:3::2
Device(config-router-lisp)# ipv6 itr
Device(config-router-lisp)# ipv6 etr map-server 2001:DB8:3::2 key R4KEY
Device(config-router-lisp)# ipv6 etr
Device(config-router-lisp)# exit
Device(config)# ipv6 route ::/0 2001:DB8:4::1

```

Example: Configuring MSMR

```

Device> enable
Device# configure terminal
Device(config)# interface ethernet0/0
Device(config-if)# ip address 10.0.2.1 255.255.255.252
Device(config-if)# ipv6 address 2001:DB8:1::1/64
Device (config-if)# router lisp
Device(config-router-lisp)# locator-set rtr-set1
Device(config-router-lisp-locator-set)# 10.0.3.1 priority 1 weight 1
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# locator-set rtr-set2
Device(config-router-lisp-locator-set)# 2001:DB8:2::1/64 priority 1 weight 1
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# locator-scope s1
Device(config-router-lisp-locator-scope)# rtr-locator-set rtr-set1
Device(config-router-lisp-locator-scope)# rloc-prefix 0.0.0.0/0
Device(config-router-lisp-locator-scope)# exit
Device(config-router-lisp)# locator-scope s2
Device(config-router-lisp-locator-scope)# rtr-locator-set rtr-set2
Device(config-router-lisp-locator-scope)# rloc-prefix ::/0
Device(config-router-lisp-locator-scope)# exit
Device(config-router-lisp)# site R4
Device(config-router-lisp-site)# authentication-key R4KEY
Device(config-router-lisp-site)# eid-prefix 10.10.10.0/24
Device(config-router-lisp-site)# eid-prefix 2001:DB8::/48

```

Example: Configuring RTR

```

Device(config-router-lisp-site)# exit
Device(config-router-lisp)# site R6
Device(config-router-lisp-site)# authentication-key R6KEY
Device(config-router-lisp-site)# eid-prefix 172.16.0.2/24
Device(config-router-lisp-site)# eid-prefix 2001:DB8::1/48
Device(config-router-lisp-site)# exit
Device(config-router-lisp)# ipv4 map-server
Device(config-router-lisp)# ipv4 map-resolver
Device(config-router-lisp)# exit
Device(config)# ip route 0.0.0.0 0.0.0.0 10.0.2.2
Device(config)# ipv6 route ::/0 2001:DB8:1::ABCD

```

Example: Configuring RTR

```

Device> enable
Device# configure terminal
Device(config)# interface Ethernet0/0
Device(config-if)# ip address 10.0.3.1 255.255.255.252
Device(config-if)# ipv6 address 2001:DB8:2::1/64
Device (config-if)# router lisp
Device(config-router-lisp)# locator-set setALL
Device(config-router-lisp-locator-set)# 10.0.3.1 priority 1 weight 1
Device(config-router-lisp-locator-set)# 2001:DB8:2::1 priority 1 weight 1
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# map-request itr-rlocs setALL
Device(config-router-lisp)# eid-table default instance-id 0
Device(config-router-lisp-eid-table)# map-cache 0.0.0.0/0 map-request
Device(config-router-lisp-eid-table)# map-cache ::/0 map-request
Device(config-router-lisp-eid-table)# exit
Device(config-router-lisp)# ipv4 map-request-source 10.0.3.1
Device(config-router-lisp)# ipv4 map-cache-limit 100000
Device(config-router-lisp)# ipv4 proxy-etr
Device(config-router-lisp)# ipv4 proxy-itr 10.0.3.1 2001:DB8:2::1
Device(config-router-lisp)# ipv4 itr map-resolver 10.0.2.1
Device(config-router-lisp)# ipv4 itr map-resolver 2001:DB8:1::1
Device(config-router-lisp)# ipv6 map-request-source 2001:DB8:2::1
Device(config-router-lisp)# ipv6 map-cache-limit 100000
Device(config-router-lisp)# ipv6 proxy-etr
Device(config-router-lisp)# ipv6 proxy-itr 2001:DB8:2::1 10.0.3.1
Device(config-router-lisp)# ipv6 itr map-resolver 10.0.2.1
Device(config-router-lisp)# ipv6 itr map-resolver 2001:DB8:1::1
Device(config-router-lisp)# exit
Device(config)# ip route 0.0.0.0 0.0.0.0 10.0.3.2
Device(config)# ipv6 route ::/0 2001:DB8:ABCD::1

```

Example: Verifying LISP Support for Disjoint RLOC Domains

Sample Output for the show ip lisp database Command

To display Locator/ID Separation Protocol (LISP) egress tunnel router (ETR) configured local IPv4 endpoint identifier (EID) prefixes and associated locator sets, use the **show ip lisp database** command in privileged EXEC mode.

```

Device# show ip lisp database
.
.
.

```

```
10.10.10.0/24, locator-set R4
Locator Pri/Wgt Source State
10.0.4.1 1/1 cfg-addr site-self, reachable
```

Sample Output for the show ipv6 lisp database Command

To display LISP ETR configured local IPv6 EID prefixes and associated locator sets, use the **show ip lisp database** command in privileged EXEC mode.

```
Device# show ipv6 lisp database
.
.
.
2001:DB8::/48, locator-set R4
Locator Pri/Wgt Source State
10.0.4.1 1/1 cfg-addr site-self, reachable
mm
```

Sample Output for the show lisp site detail Command

To display configured LISP sites on a LISP map server, use the **show lisp site detail** in privileged EXEC mode.

```
Device# show lisp site detail
.
.
.
Site name: R4
.
.
.
EID-prefix: 10.10.10.0/24
.
.
.
ETR 10.0.4.1, last registered 00:00:52, no proxy-reply, map-notify
TTL 1d00h, no merge, hash-function sha1, nonce 0x28517C31-0x7B233E66
state complete, no security-capability
xTR-ID 0xEC52ECC2-0x006CEAFE-0x814263B3-0x89675EB6
site-ID unspecified
Locator Local State Pri/Wgt Scope
10.0.4.1 yes up 1/1 s1
EID-prefix: 2001:DB8::/48
.
.
.
.
ETR 10.0.4.1, last registered 00:00:39, no proxy-reply, map-notify
TTL 1d00h, no merge, hash-function sha1, nonce 0xF91CB211-0x5B00E72C
state complete, no security-capability
xTR-ID 0xEC52ECC2-0x006CEAFE-0x814263B3-0x89675EB6
site-ID unspecified
Locator Local State Pri/Wgt Scope
10.0.4.1 yes up 1/1 s1
.
.
.
```

Sample Output for the show ip lisp map-cache Command

To display the current dynamic and static IPv4 endpoint identifier-to-routing locator (EID-to-RLOC) map cache entries, use the **show ip lisp map-cache** command in privileged EXEC mode.

```
Device# show ip lisp map-cache

LISP IPv4 Mapping Cache for EID-table default (IID 0), 2 entries
.
.
.
172.16.0.2/24, uptime: 00:01:14, expires: 00:13:44, via map-reply, complete
  Locator  Uptime   State    Pri/Wgt
  10.0.3.1  00:01:14  up       1/1
```

Sample Output for the show ipv6 lisp map-cache Command

To display the current dynamic and static IPv6 EID-to-RLOC map-cache entries, use the **show ipv6 lisp map-cache** command in privileged EXEC mode.

```
Device# show ipv6 lisp map-cache

LISP IPv6 Mapping Cache for EID-table default (IID 0), 2 entries
.
.
.
2001:DB8::1/48, uptime: 00:02:18, expires: 00:12:44, via map-reply, complete
  Locator  Uptime   State    Pri/Wgt
  10.0.3.1  00:02:18  up       1/1
```

Additional References for LISP Support for Disjoint RLOC Domains

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Locator/ID Separation Protocol (LISP) commands	Cisco IOS IP Routing: LISP Command Reference

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP Support for Disjoint RLOC Domains

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Feature Name	Release	Feature Information
LISP Support for Disjoint RLOC Domains		The LISP Support for Disjoint RLOC domains feature enables LISP-to-LISP communications between LISP sites that are connected to different RLOC spaces but have no connectivity to each other.



CHAPTER 11

LISP Data Plane Security

The Locator/ID Separation Protocol (LISP) Data Plane Security feature ensures that only traffic from within a LISP VPN can be decapsulated into the VPN. The feature is enforced when LISP packets are decapsulated by a tunnel router at the destination. Egress tunnel routers (ETRs) and proxy egress tunnel routers (PETRs) validate that the source Routing Locator (RLOC) address carried by the data packet is a member of the LISP VPN.

The solution relies on Unicast Reverse Path Forwarding (uRPF) being implemented in the RLOC network to ensure that the RLOC source addresses in LISP encapsulated data packets cannot be spoofed. Packets from outside the LISP VPN carry invalid source RLOCs that are blocked during decapsulation by ETRs and PETRs.

The advantages of implementing the LISP Data Plane Security feature are given below:

- Enhanced security due to validation by ETRs and PETRs during decapsulation.
- [Finding Feature Information, on page 213](#)
- [Prerequisites for LISP Data Plane Security, on page 214](#)
- [Restrictions for LISP Data Plane Security, on page 214](#)
- [Information About LISP Data Plane Security, on page 214](#)
- [How to Configure LISP Data Plane Security, on page 216](#)
- [Configuration Examples for LISP Data Plane Security, on page 223](#)
- [Additional References for LISP Data Plane Security, on page 224](#)
- [Feature Information for LISP Data Plane Security, on page 225](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for LISP Data Plane Security

- Understanding of LISP concepts, including the concept of virtual routing and forwarding (VRF) instances bound to instance IDs (IIDs). These concepts are explained in the chapters *LISP Overview*, *Configuring LISP*, and *LISP Shared Model Virtualization*.
- uRPF implementation in the RLOC network.

Restrictions for LISP Data Plane Security

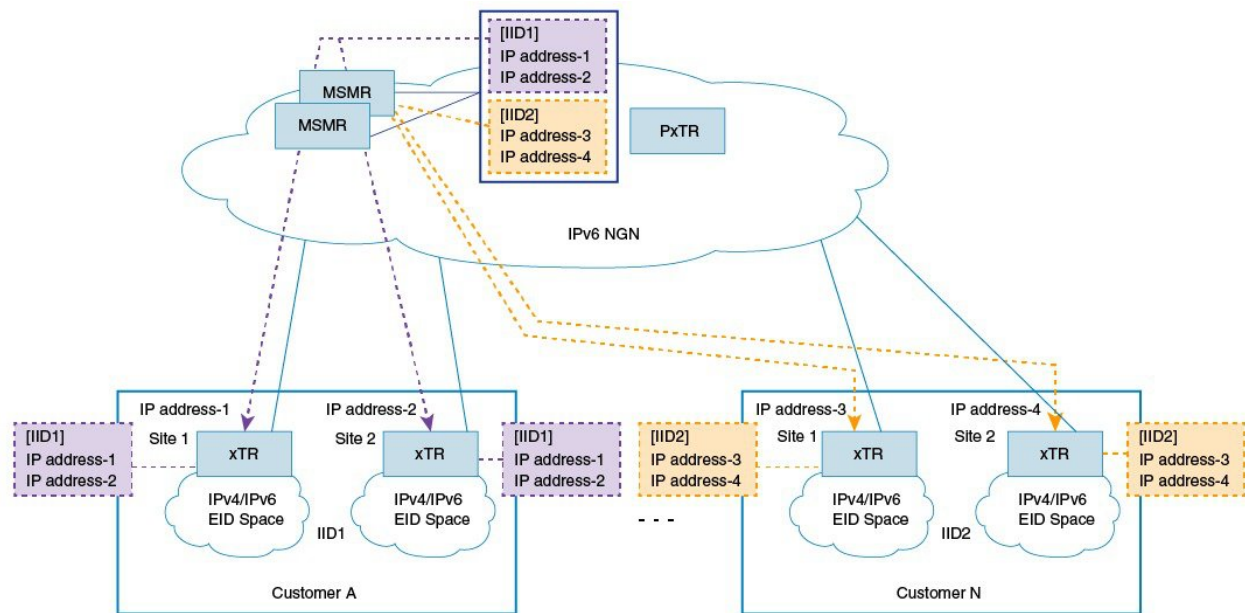
- All sites within a given LISP VPN must register to one or a common set of Map-Servers. That is, all IP prefixes associated with a specific instance ID must be delegated from common a Map-Server to ensure that these Map-Servers can construct a complete RLOC set for the given LISP VPN.

Information About LISP Data Plane Security

Source RLOC Decapsulation Filtering

This feature enhances data plane security by monitoring LISP packets during the decapsulation stage, when the packets are sent from an ingress tunnel router (ITR) or proxy ingress tunnel router (PITR) to an ETR or PETR. To protect LISP VPN end sites from decapsulating LISP packets that do not belong to the VPN, whether the result of misconfiguration or an attack, the source address in the incoming LISP packets are compared with a dynamically distributed set of source RLOC addresses corresponding to valid LISP VPN end sites. LISP packet decapsulation by ETRs and PETRs validate that a source RLOC address of an incoming LISP data packet is a member of the VPN. Note that this solution requires that source RLOC addresses are not spoofed, and hence unicast RPF or ingress anti-spoofing access control lists (ACLs) are required within the RLOC core network.

Consider the scenario in the image below:



1. Customer A has 2 LISP sites, site1 and site2, each having an xTR (a device performing the role of ETR and ITR). Site 1 and Site 2 register with the Map-Servers (of the Map-Server/Map-Resolver [MSMR] devices) supporting the LISP control plane for the LISP VPN with instance ID 1. The Map-Server automatically records the registration RLOCs for both sites, and dynamically pushes this list of valid RLOCs to both sites. In this way, site 1 and site 2 of the customer A LISP VPN can send traffic between each other. No other LISP encapsulated traffic is permitted, as the source RLOC will not match the valid source RLOC list.
2. Customer N also has 2 LISP sites, site1 and site2, and both register to the Map-Servers supporting the LISP control plane for this LISP VPN with instance ID 2. The Map-Server automatically records the registration RLOCs for both sites, and dynamically pushes this list of valid RLOCs to both sites. In this way, site 1 and site 2 of the customer N LISP VPN can send traffic between each other. No other LISP encapsulated traffic is permitted, as the source RLOC will not match the valid source RLOC list.

In addition to the automatically learned source RLOCs of registering LISP sites, the per-IID (instance ID) membership list can be extended to include specific source RLOCs of valid devices that do not register, such as PxTRs. When this feature is deployed, the source RLOCs of the PxTR is made available with the xTRs.

Some pointers for implementing source RLOC decapsulation filtering are given below :

- For Map-Servers to be able to construct the complete list of members for an EID instance ID, they must receive registrations from all the xTRs participating in the customer VPN.
- Map-Servers construct the EID instance ID-RLOC membership list using the RLOC information in the received mapping records in map-register messages.
- All IP prefixes associated with a specific instance ID must be delegated from a common Map-Server to ensure that these Map-Servers can construct a complete RLOC set for the given LISP VPN.
- All xTRs within a VPN must register with a common set of Map-Servers.

- PxTRs do not (normally) register with the Map-Servers, such that the Map-Servers could discover the PxTR RLOC, and that the Map-Servers could distribute learned RLOCs to the PxTRs. Thus, PxTR RLOCs need to be manually configured on the Map-Server.
- The EID instance membership lists built by Map-Servers are communicated only to xTRs and PxTRs that are members of the VPN.

TCP-based Sessions for LISP Packet Transport

The LISP data plane security mechanism requires the automated distribution and updating of RLOC filter lists to VPN members. This automated distribution is accomplished through a TCP-based session established between the xTRs and Map-Servers after the normal registration process has completed.

For example, xTRs periodically transmit map register messages and process the resulting map notify messages issued by the Map-Server. The Map-Servers process map register messages, update corresponding registration state, and transmit matching map notify messages.

To implement a more reliable, secure, and scalable transport option, TCP-based sessions are provided for LISP-related communication between xTRs and Map-Servers.

Some pointers regarding TCP-based sessions are given below:

- The UDP-based registration mechanism is conducted, and then a TCP-based session is established and used for the distribution of EID-instance RLOC membership lists. The number of xTRs that a Map-Server can support is limited by the number of TCP sessions that the Map-Server can establish and maintain. This determines the number of VPN customers that a Map-Server can host.
- The xTRs belonging to the same VPN must register with the same Map-Servers. This limits the number of sites within a VPN to the number of TCP sessions that a Map-Server can support.

How to Configure LISP Data Plane Security

Configuring MSMR

To configure the MSMR devices, perform the steps given below:



Note Steps 5 to 10 are optional. You can use those to modify the list of RLOC addresses (filter list) discovered by the Map-Server.

Before you begin

- Ensure that you have available any RLOCs associated with PxTRs serving within the LISP VPN.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router lisp**

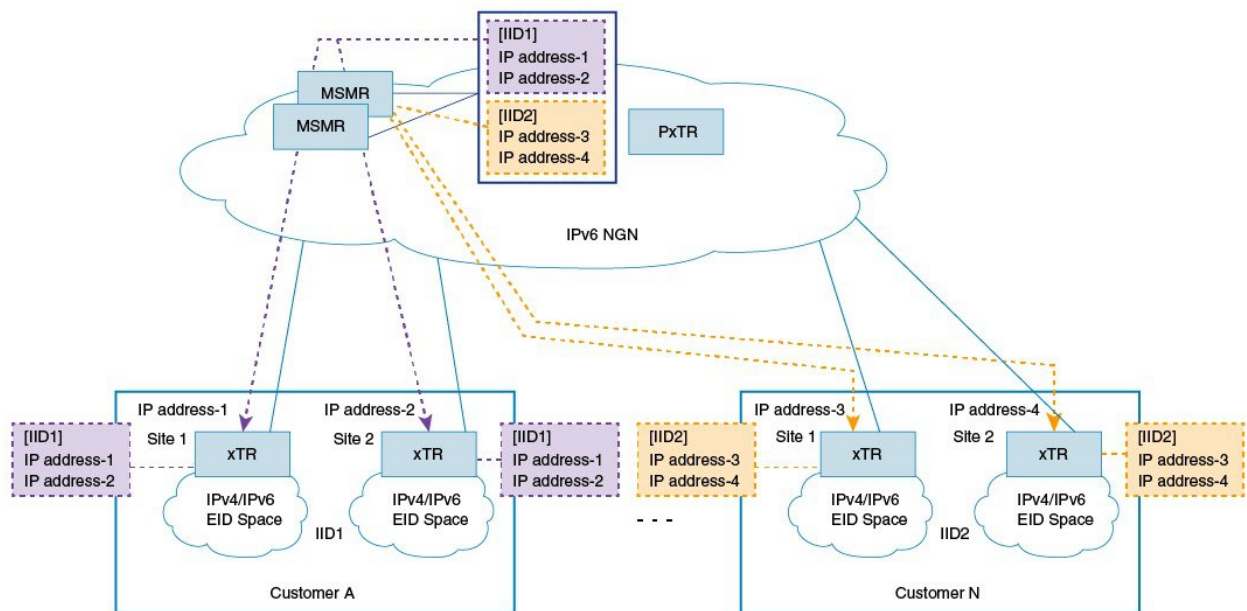
4. **map-server rloc members distribute**
5. **locator-set locator-set-name**
6. *ipv4-address* **priority value weight value**
7. **exit**
8. **eid-table vrf vrf-name instance-id iid**
9. **map-server rloc members modify-discovered add locator-set locator-set-name**
10. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router lisp Example: Device(config)# router lisp	Enters LISP configuration mode.
Step 4	map-server rloc members distribute Example: Device(config-router-lisp)# map-server rloc members distribute	Enables distribution of the list of EID prefixes to xTRs at the customer end.
Step 5	locator-set locator-set-name Example: Device(config-router-lisp)# locator-set PTR_set	(Optional) Specifies a locator set for the PxTR and enters LISP locator set configuration mode.
Step 6	<i>ipv4-address</i> priority value weight value Example: Device(config-router-lisp-locator-set)# 10.10.10.1 priority 1 weight 1	(Optional) Configures the LISP locator set. You can configure each locator address by creating a locator entry with an assigned priority and weight
Step 7	exit Example: Device(config-router-lisp-locator-set)# exit	(Optional) Exits LISP locator set configuration mode and enters LISP configuration mode.
Step 8	eid-table vrf vrf-name instance-id iid Example: Device(config-router-lisp)# eid-table vrf cust-A instance-id 1	(Optional) Configures an association between a VRF table and a LISP instance ID, and enters eid-table configuration submode.

	Command or Action	Purpose
Step 9	map-server rloc members modify-discovered add locator-set locator-set-name Example: <pre>Device(config-router-lisp-eid-table)# map-server rloc members modify-discovered add locator-set PTR_set</pre>	(Optional) Adds RLOC addresses in the specified locator set to the list of <i>discovered</i> RLOC addresses. Note The updated list will be sent to the xTRs at the customer end when the distribution option is enabled.
Step 10	exit Example: <pre>Device(config-router-lisp-eid-table)# exit</pre>	(Optional) Exits eid-table configuration submode and enters LISP configuration mode.

Configuring the xTRs



To enable data plane security on the xTRs belonging to customer A (as shown in the image), configure the xTR at site1, as shown below:

Before you begin

- Ensure that you have configured the MSMR devices.
- Ensure that uRPF is implemented in the RLOC network.
- Ensure that you have identified EIDs and the LISP device acting as an xTR.

SUMMARY STEPS

1. enable
2. configure terminal

3. **router lisp**
4. **decapsulation filter rloc source member**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router lisp Example: Device(config)# router lisp	Enters LISP configuration mode.
Step 4	decapsulation filter rloc source member Example: Device(config-router-lisp)# decapsulation filter rloc source member	Enables source RLOC address validation of LISP packets.
Step 5	exit Example: Device(config-router-lisp)# exit	Exits LISP configuration mode and returns to global configuration mode.

What to do next

- The above steps enable data plane security for the xTR at one of customer A's sites, 'site1'. You need to repeat the steps to enable RLOC decapsulation filtering for customer A's second site, 'site2'.

Configuring PxTR

To configure the PxTR, perform the steps given below:

Before you begin

- Ensure that the MSMR devices and xTRs at the customer sites are configured.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router lisp**

4. decapsulation filter rloc source members
5. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router lisp Example: Device(config)# router lisp	Enters LISP configuration mode.
Step 4	decapsulation filter rloc source members Example: Device(config-router-lisp)# decapsulation filter rloc source members	Enables source RLOC address validation of LISP packets.
Step 5	exit Example: Device(config-router-lisp)# exit	Exits LISP configuration mode and returns to global configuration mode.

What to do next

- Configure any other PxTR as needed.

Verifying LISP Data Plane Security On a Map-Server

Verify the LISP Data Plane Security feature on a Map-Server by using the commands given below:

SUMMARY STEPS

1. show lisp [session [established] | vrf [vrf-name [session [peer-address]]]]
2. show lisp site rloc members [instance-id iid]

DETAILED STEPS

Step 1 show lisp [session [established] | vrf [vrf-name [session [peer-address]]]]

Example:

```
Device# show lisp session
```

```
Sessions for VRF default, total: 8, established: 7
Peer                State    Up/Down    In/Out    Users
2001:DB8:A:1::2    Up       00:04:13   2/7       2
2001:DB8:A:2::2    Up       00:04:13   2/7       2
2001:DB8:A:3::2    Up       00:03:53   2/7       2
2001:DB8:B:1::2    Up       00:04:04   2/6       2
2001:DB8:B:2::2    Init     never      0/0       1
2001:DB8:C:1::2    Up       00:03:55   2/6       2
2001:DB8:C:2::2    Up       00:03:54   2/6       2
2001:DB8:E:F::2    Up       00:04:04   6/19      4
```

This command displays reliable transport session information. If there is more than one transport session, the corresponding information will be displayed.

Step 2 `show lisp site rloc members [instance-id iid]`

Example:

```
Device# show lisp site rloc members

LISP RLOC membership for EID table default (IID 0), 5 entries

RLOC                Origin                Valid
10.0.1.2            registration          Yes
10.0.2.2            config & registration Yes
```

The **Origin** column displays configuration details of the RLOC member – whether the RLOC member is manually configured, automatically gleaned from received registrations, or both. The **Valid** column shows whether the RLOC is a valid member that is distributed to (P)xTRs. A listed RLOC may not be valid if it is gleaned from registrations but the 'override' option is used in the 'modify-discovered' configuration, and the specified locator-set does not include the RLOC.

Verifying and Troubleshooting LISP Data Plane Security on an xTR or PxTR

Verify the LISP Data Plane Security feature on an xTR or PxTR by using the commands given below:

SUMMARY STEPS

1. `show lisp [session [established] | vrf [vrf-name [session [peer-address]]]]`
2. `show lisp decapsulation filter [IPv4-rloc-address | IPv6-rloc-address] [eid-table eid-table-vrf | instance-id iid]`
3. `show cef source-filter table`
4. `debug lisp control-plane eid-membership`
5. `debug lisp control-plane session`

DETAILED STEPS

Step 1 `show lisp [session [established] | vrf [vrf-name [session [peer-address]]]]`

Example:

```
Device# show lisp session

Sessions for VRF default, total: 8, established: 7
Peer                State    Up/Down    In/Out    Users
```

2001:DB8:A:1::2	Up	00:04:13	2/7	2
2001:DB8:A:2::2	Up	00:04:13	2/7	2
2001:DB8:A:3::2	Up	00:03:53	2/7	2
2001:DB8:B:1::2	Up	00:04:04	2/6	2
2001:DB8:B:2::2	Init	never	0/0	1
2001:DB8:C:1::2	Up	00:03:55	2/6	2
2001:DB8:C:2::2	Up	00:03:54	2/6	2
2001:DB8:E:F::2	Up	00:04:04	6/19	4

This command displays reliable transport session information. If there is more than one transport session, the corresponding information will be displayed.

Step 2 **show lisp decapsulation filter** [*IPv4-rloc-address* | *IPv6-rloc-address*] [**eid-table** *eid-table-vrf* | **instance-id** *iid*]

Example:

```
Device# show lisp decapsulation filter instance-id 0

LISP decapsulation filter for EID-table default (IID 0), 3 entries

Source RLOC      Added by
10.0.0.1         Config
10.0.0.5         209.165.200.230 209.165.200.232
10.0.0.6         Config 209.165.200.230
```

The RLOC address configuration details (whether it is manually configured or discovered) on a (P)xTR is displayed in the above table.

Step 3 **show cef source-filter table**

Example:

```
Device# show cef source-filter table

[lisp:0:0:IPv4] state [enabled, active], 0 entries, refcount 3, flags [], action [drop]
Database epoch 0
Hits 0, misses 0, fwd 0, drop 0
```

This command displays Cisco Express Forwarding (CEF) source-filter tables.

Step 4 **debug lisp control-plane eid-membership**

Example:

```
Device# debug lisp control-plane eid-membership

LISP control plane EID membership debugging is on
```

Displays debugging information for EID membership discovery.

Step 5 **debug lisp control-plane session**

Example:

```
Device# debug lisp control-plane session

LISP control plane session debugging is on
```


Displays detailed session establishment debugging information.

Configuration Examples for LISP Data Plane Security

Example: Configuring MSMR



Note Steps for adding the locator set and the RLOC address are optional. You can use those steps to modify the list of RLOC addresses (filter list) discovered by the Map-Server.

```
Device> enable
Device# configure terminal
Device(config)# router lisp
Device(config-router-lisp)# map-server rloc members distribute
Device(config-router-lisp)# locator-set PTR_set
Device(config-router-lisp-locator-set)# 10.10.10.1 priority 1 weight 1
Device(config-router-lisp-locator-set)# exit
Device(config-router-lisp)# eid-table vrf cust-A instance-id 1
Device(config-router-lisp-eid-table)# map-server rloc members modify-discovered add
locator-set PTR_set
Device(config-router-lisp-eid-table)# exit
```

Repeat the above steps to configure one or more map servers, as needed

Example: Configuring the xTRs

```
Device> enable
Device# configure terminal
Device(config)# router lisp
Device(config-router-lisp)# decapsulation filter rloc source member
Device(config-router-lisp)# exit
```

The above steps enable data plane security for the xTR at one of customer sites. You must repeat the steps to enable RLOC decapsulation filtering for other sites.

Example: Configuring PxTR

```
Device> enable
Device# configure terminal
Device(config)# router lisp
Device(config-router-lisp)# decapsulation filter rloc source member
```

```
Device(config-router-lisp)# exit
```

Additional References for LISP Data Plane Security

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Locator/ID Separation Protocol (LISP) commands	Cisco IOS IP Routing: LISP Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 6830	<i>Locator/ID Separation Protocol (LISP)</i>
RFC 6832	<i>Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites</i>
RFC 6833	<i>Locator/ID Separation Protocol (LISP) Map-Server Interface</i>

MIBs

MIB	MIBs Link
• CRUMB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP Data Plane Security

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 6: Feature Information for LISP Data Plane Security

Feature Name	Releases	Feature Information
LISP Data Plane Security		<p>The LISP Data Plane Security feature ensures that only traffic from within a LISP VPN can be decapsulated into the VPN.</p> <p>The following commands were introduced by this feature: clear lisp vrf, decapsulation filter rloc source, debug lisp control-plane eid-membership, debug lisp control-plane session, map-server rloc members distribute, map-server rloc members modify-discovered, show lisp decapsulation filter, show lisp site rloc members, show lisp session</p>



CHAPTER 12

LISP Reliable Registration

The LISP Reliable Registration feature supports establishment of TCP based reliable map registration between Egress Tunnel Router (ETR) and Map Server (MS).

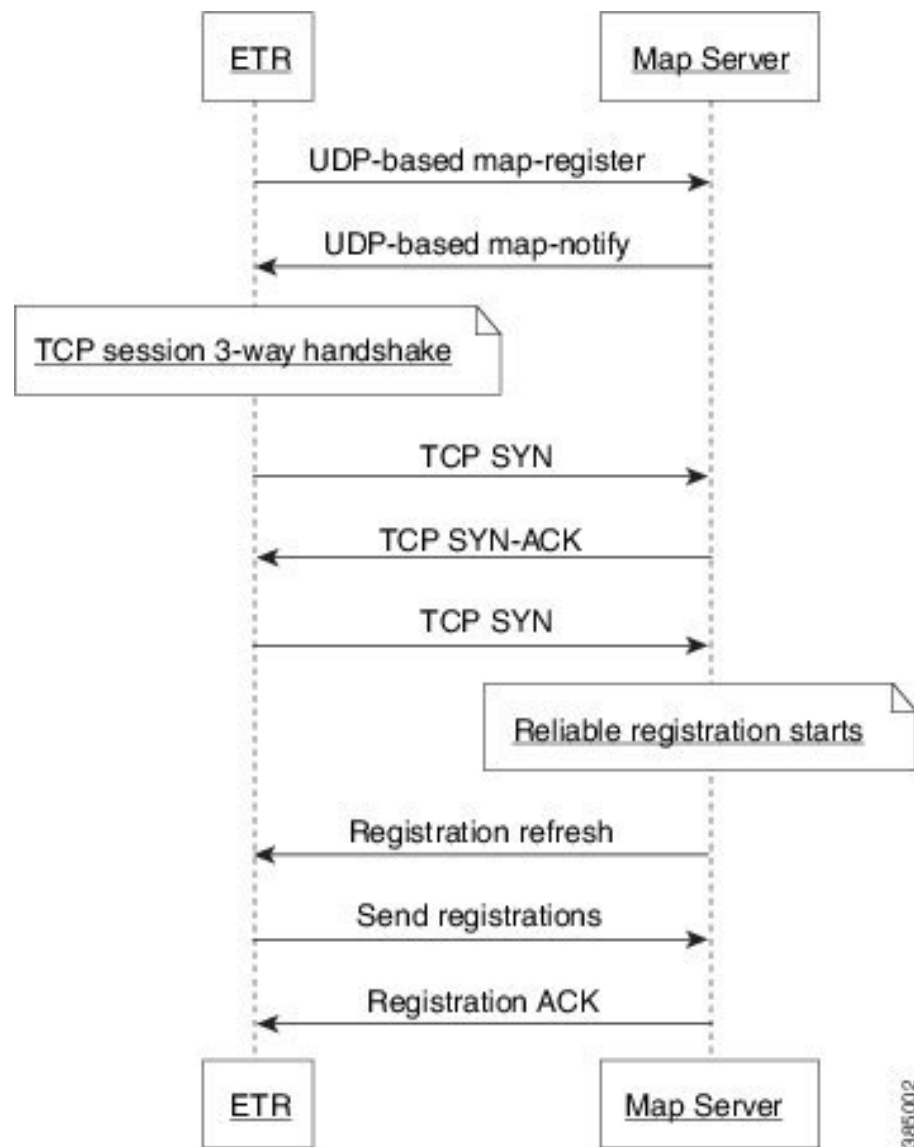
- [Information About LISP Reliable Registration, on page 227](#)
- [Additional References for LISP Reliable Registration, on page 231](#)
- [Feature Information for LISP Reliable Registration, on page 231](#)

Information About LISP Reliable Registration

LISP Reliable Map Registration

LISP ETR periodically sends UDP based map registration message to map server. This results in control traffic and scalability problems. TCP based reliable map registration or LISP reliable map registration mechanism is developed as an enhancement and replacement to the UDP based map registration mechanism.

Figure 47: LISP Reliable Map Registration Mechanism



The LISP reliable map registration mechanism as shown in the figure is described below:

- ETR sends UDP based map registration message to map server.
- Map server processes map registration and sends map-notify to ETR. This message serves as acknowledgment.
- ETR initiates a TCP session with map-server using three-way handshake.



Note When TCP based map registration is not supported by map server then ETR uses UDP based map registration to establish a session with the map server.

- Once the TCP session is established, map-server sends a registration refresh message to the ETR.

- ETR sends map registrations to the map server through the TCP connection.
- Map server acknowledges for the map registrations.

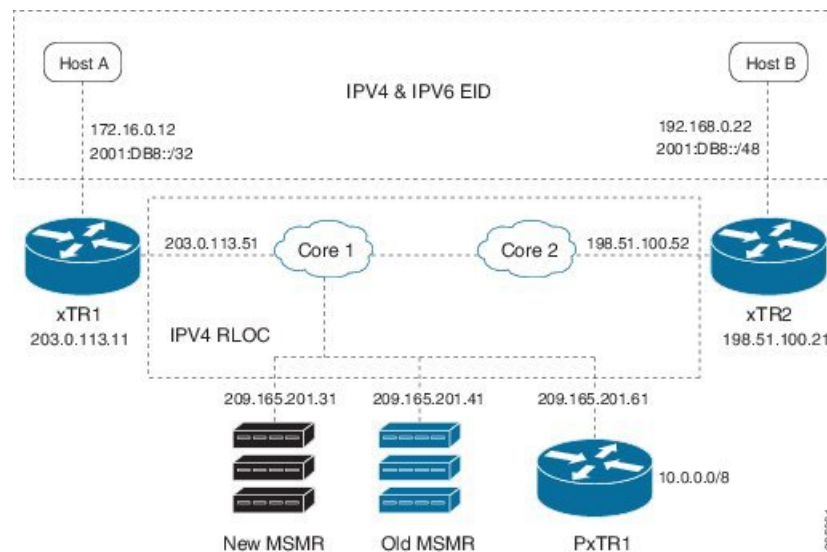


Note There are no configuration commands for this feature. This feature is turned on automatically.

Verifying the LISP Reliable Registration

Perform this task to verify the LISP Reliable Registration feature which is enabled automatically in the LISP network. In this example, a LISP site uses a single edge router that functions as both ITR and ETR (known as an xTR). Routing Locators (RLOCs) are in IPv4. EID prefixes are in both IPv4 and IPv6. The LISP site registers to two map server/map resolver (MSMR) devices in the network core. The topology used in verifying LISP Reliable Registration is as shown in the figure below.

Figure 48: LISP Reliable Registration Topology



The components as shown in the topology are described below:

- xTR1 and xTR2 are xTRs for 2 LISP sites.
- Core1 and Core 2 are routing locators (RLOCs) core routers with no LISP configuration.
- New MSMR is a map-server and map-resolver with reliable map-registration support, whereas Old MSMR does not support reliable map-registration.
- PxTR1 works as a Proxy Ingress Tunnel Router (PITR) and Proxy Egress Tunnel Router (PETR) between the network with 10.0.0.0/8 prefix and the LISP sites.
- Only static routing protocols are used in this setup to reduce control traffic.

In the following output, a '#' sign in the 'Up' column indicates reliable map registration session.

```
Device# show lisp site
```

```
LISP Site Registration Information
* = Some locators are down or unreachable
# = Some registrations are sourced by reliable transport
```

Site Name	Last Register	Up	Who Last Registered	Inst ID	EID Prefix
A	never	no	--		0.0.0.0/0
	01:59:44	yes#	203.0.113.11		10.10.10.0/24
	01:59:44	yes#	203.0.113.11		10.20.20.0/24
	01:59:44	yes#	203.0.113.11		172.16.0.0/24
	01:59:44	yes#	203.0.113.11		2001:DB8::/32
B	never	no	--		0.0.0.0/0
	never	no	--		10.0.0.0/8
	01:59:43	yes#	198.51.100.21		10.30.30.0/24
	01:59:43	yes#	198.51.100.21		10.40.40.0/24
	never	no	--		21.0.0.0/8
	01:59:43	yes#	198.51.100.21		21.21.21.0/24
	01:59:43	yes#	198.51.100.21		2001:DB8::/48

In the following output, no '#' sign in the 'Up' column indicates that the Old MSMR does not support reliable map registration.

```
Device# show lisp site
```

```
LISP Site Registration Information
* = Some locators are down or unreachable
```

Site Name	Last Register	Up	Who Last Registered	Inst ID	EID Prefix
A	never	no	--		0.0.0.0/0
	00:00:00	yes	203.0.113.11		172.16.0.0/24
	00:00:55	yes	198.51.100.21		21.21.21.0/24
	00:00:03	yes	203.0.113.11		2001:DB8::/32
B	never	no	--		10.0.0.0/8
	00:00:00	yes	203.0.113.11		10.10.10.0/24
	00:00:00	yes	203.0.113.11		10.20.20.0/24
	00:00:55	yes	198.51.100.21		10.30.30.0/24
	00:00:55	yes	198.51.100.21		10.40.40.0/24
	00:00:52	yes	198.51.100.21		2001:DB8::/48

The following output is from xTR1 that uses 2 map servers. Reliable map-registration session is established with 209.165.201.31 (New MSMR), but not with 209.165.201.41 (Old MSMR).

```
Device# show lisp session
```

```
Sessions for VRF default, total: 2, established: 1
Peer                State    Up/Down    In/Out    Users
209.165.201.31     Up       05:05:40   6/3       2
209.165.201.41     Down     never      0/0       1
```

The following output is from New MSMR. It has established reliable map-registration sessions with two ETRs.

```
Device# show lisp session
```

```
Sessions for VRF default, total: 2, established: 2
Peer                State    Up/Down    In/Out    Users
203.0.113.11       Up       05:19:53   3/6       1
198.51.100.21      Up       05:18:28   2/5       1
```


Additional References for LISP Reliable Registration

Related Documents

Document Title	Location
Cisco IOS commands	Cisco IOS Master Command List, All Releases
LISP commands	Cisco IOS IP Routing: LISP Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 6830	<i>The Locator/ID Separation Protocol (LISP)</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP Reliable Registration

Table 7: Feature Information for LISP Reliable Registration

Feature Name	Releases	Feature Information
LISP Reliable Registration		The LISP Reliable Registration feature supports establishment of TCP based reliable map-registration between Egress Tunnel Router (ETR) and Map Server (MS). The following commands were modified: show lisp site .



CHAPTER 13

Overlapping Prefix

The Overlapping prefix feature supports Endpoint Identifier (EID) registration by two sites where the EID prefix from one LISP site is a subset of the EID prefix from another LISP site.

- [Prerequisites for Overlapping Prefix, on page 233](#)
- [Information About Overlapping Prefix, on page 233](#)
- [How to Configure Overlapping Prefix, on page 234](#)
- [Additional References for Overlapping Prefix, on page 235](#)
- [Feature Information for Overlapping Prefix, on page 236](#)

Prerequisites for Overlapping Prefix

- Reliable registration must be established between the xTR (performs functions of both Egress Tunnel Router and Ingress Tunnel Router components) and map server/map resolver (MS/MR).

Information About Overlapping Prefix

Endpoint ID (EID)

An EID value for IPv4 is 32 bit and EID value for IPv6 is 128-bit. EIDs are used in the source and destination address fields of the first LISP header of a packet.

EID-Prefix

An EID-Prefix is a power-of-two blocks of EIDs allocated to a LISP site by an address allocation authority.

Map Server/Map Resolver (MS/MR)

MS and MR functions are implemented on the same device, which is referred to as an MS/MR device.

How to Configure Overlapping Prefix

Configuring Overlapping Prefix

Configure EID-prefix with "accept-more-specifices" keyword to allow MS to accept registration of more specific prefix.

```
router lisp
  site site3
    authentication-key cisco
    eid-prefix 172.16.0.0/8 accept-more-specifices
  exit
```

Register 3.0.0.0/8 with MS.

```
router lisp
  database-mapping 172.16.0.0/8 10.0.0.3 priority 1 weight 100
```

Register 3.1.0.0/16 with MS, which is more specific and overlap with 3.0.0.0/8 prefix registered from xTR3.

```
router lisp
  database-mapping 192.168.0.0/16 10.0.0.4 priority 1 weight 100
  database-mapping 192.0.2.0/8 10.0.0.4 priority 1 weight 100
```

Verifying Overlapping Prefix

Perform this task to verify the Overlapping Prefix feature in the LISP network. In this example, there are four routers: MSMR, xTR2, xTR3, and xTR4. Each router has an interface connection in the same subnet (RLOC space) 10.0.0.0/24. The following are the IP addresses of the routers:

Router	IP Address
MSMR	10.0.0.1
xTR2	10.0.0.2
xTR3	10.0.0.3
xTR4	10.0.0.4

MS/MR Output:

```
Device# show lisp site
```

```
LISP Site Registration Information
```

```
* = Some locators are down or unreachable
```

```
# = Some registrations are sourced by reliable transport
```

Site Name	Last Register	Up	Who Registered	Inst ID	EID Prefix
site2	00:15:08	yes#	10.0.0.2		2.0.0.0/8
site3	00:15:05	yes#	10.0.0.3		3.0.0.0/8
	00:15:01	yes#	10.0.0.4		3.1.0.0/16
site4	00:15:01	yes#	10.0.0.4		4.0.0.0/8

xTR1 Output:

```
Device# show ip lisp map-cache

LISP IPv4 Mapping Cache for EID-table default (IID 0), 3 entries

0.0.0.0/0, uptime: 00:18:05, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
3.0.0.0/8, uptime: 00:00:16, expires: 23:59:43, via map-reply, complete
  Locator  Uptime   State   Pri/Wgt
  10.0.0.3 00:00:16 up      1/100
3.1.0.0/16, uptime: 00:00:08, expires: 23:59:51, via map-reply, complete
  Locator  Uptime   State   Pri/Wgt
  10.0.0.4 00:00:08 up      1/100
```

xTR2 Output:

```
Device# show ip lisp map-cache

LISP IPv4 Mapping Cache for EID-table default (IID 0), 3 entries

0.0.0.0/0, uptime: 00:18:44, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
2.0.0.0/8, uptime: 00:00:57, expires: 23:59:02, via map-reply, complete
  Locator  Uptime   State   Pri/Wgt
  10.0.0.2 00:00:57 up      1/100
3.1.0.0/16, uptime: 00:18:40, expires: 23:42:12, via map-reply, self, complete
  Locator  Uptime   State   Pri/Wgt
  10.0.0.4 00:17:47 up      1/100
```

```
Device# show ip lisp away
```

```
LISP Away Table for router lisp 0 (default) IID 0
Entries: 1
```

```
Prefix                               Producer
3.1.0.0/16                            mapping-notification
```

xTR3 Output:

```
Device# show ip lisp map-cache

LISP IPv4 Mapping Cache for EID-table default (IID 0), 2 entries

0.0.0.0/0, uptime: 00:19:26, expires: never, via static send map-request
  Negative cache entry, action: send-map-request
2.0.0.0/8, uptime: 00:01:35, expires: 23:58:24, via map-reply, complete
  Locator  Uptime   State   Pri/Wgt
  10.0.0.2 00:01:35 up      1/100
```

```
Device# show ip lisp away
```

```
LISP Away Table for router lisp 0 (default) IID 0
Entries: 0
```

Additional References for Overlapping Prefix

Related Documents

Document Title	Location
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Document Title	Location
LISP commands	Cisco IOS IP Routing: LISP Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 6830	<i>The Locator/ID Separation Protocol (LISP)</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Overlapping Prefix

Table 8: Feature Information for Overlapping Prefix

Feature Name	Releases	Feature Information
Overlapping Prefix		<p>The Overlapping prefix feature supports Endpoint Identifier (EID) registration by two sites where the EID prefix from one LISP site is a subset of the EID prefix from another LISP site.</p> <p>The following commands were modified: authentication-key, database-mapping, router lisp.</p>



CHAPTER 14

LISP Generalized SMR

The LISP Generalized SMR feature enables LISP xTR (ITR and ETR) to update map cache when there is a change in database mapping.



Note There is no configuration commands for this feature. This feature is turned on automatically.

- [Information About LISP Generalized SMR, on page 237](#)
- [Verifying LISP Generalized SMR , on page 238](#)
- [Additional References for LISP Reliable Registration, on page 240](#)
- [Feature Information for LISP Generalized SMR, on page 241](#)

Information About LISP Generalized SMR

Solicit-Map-Request (SMR)

Soliciting a Map-Request enables ETRs to control requests for Map-Reply messages when there is change in database mapping. SMRs enable remote ITRs to update the database mappings that are cached. An SMR message is simply a bit set in a Map-Request message. An ITR or PITR will send a Map-Request when they receive an SMR message.



Note There is no configuration commands for this feature. This feature is turned on automatically.

Generalized SMR (GSMR)

SMR was mainly used to support LISP mobility. This mechanism has been generalized (Generalized Solicit Map Request - GSMR) to support the following use cases:

- De-configured local EID
- Local EID no-route (when an ETR decapsulates a data packet and finds no route for a configured local EID)

- Mobility host move out and detection
- Overlapping prefix

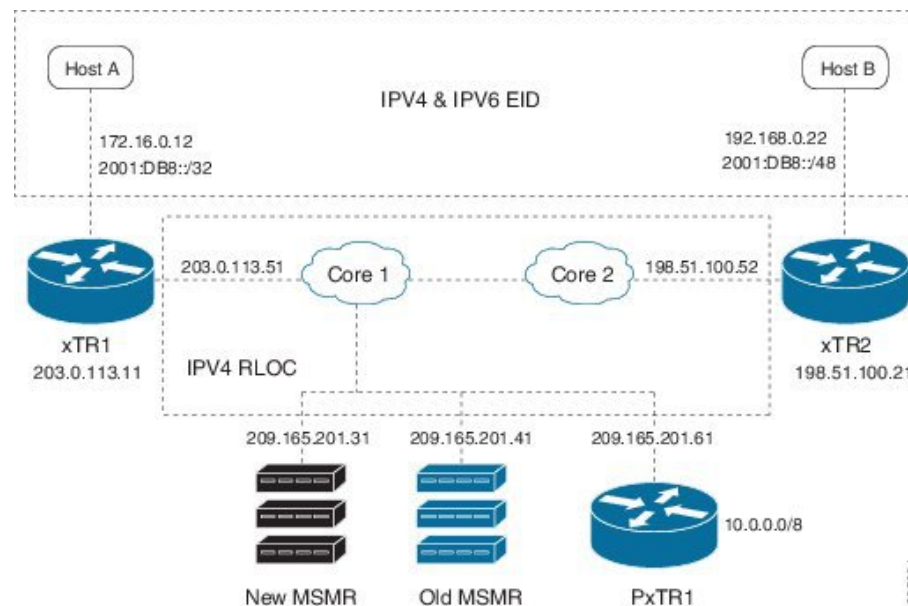


Note There are no configuration commands for this feature. This feature is turned on automatically.

Verifying LISP Generalized SMR

Perform this task to verify the LISP Generalized SMR feature which is enabled automatically in the LISP network. In this example, a LISP site uses a single edge router that functions as both ITR and ETR (known as an xTR). Routing Locators (RLOCs) are in IPv4. EID prefixes are in both IPv4 and IPv6. The LISP site registers to two map server/map resolver (MSMR) devices in the network core. The topology used in verifying LISP Generalized SMR is as shown in the figure below.

Figure 49: LISP Generalized SMR Topology



The components as shown in the topology are described below:

- xTR1 and xTR2 are xTRs for 2 LISP sites.
- Core1 and Core 2 are routing locators (RLOCs) core routers with no LISP configuration.
- New MSMR is a map-server and map-resolver with reliable map-registration support, whereas Old MSMR does not support reliable map-registration.
- PxTR1 works as a Proxy Ingress Tunnel Router (PITR) and Proxy Egress Tunnel Router (PETR) between the network with 10.0.0.0/8 prefix and the LISP sites.
- Only static routing protocols are used in this setup to reduce control traffic.

Verifying 172.16.0.0/24 is in map cache on xTR2:

```
Device# show ip lisp map-cache
LISP IPv4 Mapping Cache for EID-table default (IID 0), 3 entries
```



```

0.0.0.0/0, uptime: 03:32:45, expires: never, via static send map-request
Negative cache entry, action: send-map-request
10.20.20.0/24, uptime: 00:00:05, expires: 23:59:54, via map-reply, complete
Locator      Uptime      State      Pri/Wgt
203.0.113.11 00:00:05 up          1/100
172.16.0.0/24, uptime: 00:35:49, expires: 23:24:10, via map-reply, complete
Locator      Uptime      State      Pri/Wgt
203.0.113.11 00:35:49 up          1/100

```

Shutting down interface Ethernet1/0 on xTR1:

```

Device(config)# interface ethernet 1/0
Device(config-if)# shutdown

```

Verifying 172.16.0.0/24 is in map cache on xTR1:

```

Device# show ip lisp data
LISP ETR IPv4 Mapping Database for EID-table default (IID 0), LSBs: 0x1
Entries total 3, no-route 1, inactive 0
10.10.10.0/24, locator-set set1
Locator      Pri/Wgt Source      State
203.0.113.11 1/100  cfg-addr  site-self, reachable
10.20.20.0/24, locator-set set1
Locator      Pri/Wgt Source      State
203.0.113.11 1/100  cfg-addr  site-self, reachable
172.16.0.0/24, locator-set set1 *** NO ROUTE TO EID PREFIX ***
Locator      Pri/Wgt Source      State
203.0.113.11 1/100  cfg-addr  site-self, reachable

```

Pinging Host A from Host B:

```

Device# ping 172.16.0.12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.12, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

```

xTR1 decapsulates the data packets, finds out the no-route situation, and sends an SMR to xTR2:

```

Device#
*Feb 19 22:08:15.160: LISP: Send map request type dyn-EID SMR
*Feb 19 22:08:15.160: LISP: Send map request for EID prefix IID 0 192.168.0.22/32
*Feb 19 22:08:15.160: LISP-0: AF IID 0 IPv4, Send SMR map-request for 172.16.0.12 to
198.51.100.21.
*Feb 19 22:08:15.160: LISP-0: EID-AF IPv4, Sending probe map-request from 203.0.113.11 to
198.51.100.21
for EID 21.21.21.22/32, ITR-RLOCs 1, nonce 0x68E45971-0xE3DF4931, SMR 172.16.0.12, DoNotReply.

```

xTR2 processes the SMR and sends out a map-request to the map server:

```

Device#
*Feb 19 22:08:15.161: LISP: Processing received Map-Request(1) message on Ethernet0/0 from
203.0.113.11:4342 to 198.51.100.21:4342
*Feb 19 22:08:15.161: LISP: Received map request for IID 0 192.168.0.22/32, source_eid IID
0 172.16.0.12, ITR-RLOCs: 203.0.113.11,
records 1, nonce 0x68E45971-0xE3DF4931, probe, SMR, DoNotReply
*Feb 19 22:08:15.161: LISP-0: AF IID 0 IPv4, Scheduling SMR trigger Map-Request for
172.16.0.12/32 from 192.168.0.22.
*Feb 19 22:08:15.161: LISP-0: IID 0 SMR & D bit set, not replying to map-request.
*Feb 19 22:08:15.290: LISP: Send map request type SMR
*Feb 19 22:08:15.290: LISP: Send map request for EID prefix IID 0 172.16.0.12/32
Device#
*Feb 19 22:08:15.290: LISP-0: AF IID 0 IPv4, Send SMR triggered map request for 172.16.0.12/32
(1) from 192.168.0.22.
*Feb 19 22:08:15.290: LISP-0: EID-AF IPv4, Sending map-request from 172.16.0.12 to 172.16.0.12
for EID 172.16.0.12/32, ITR-RLOCs 1,
nonce 0x4D04AB2F-0x99FF6FF5 (encap src 198.51.100.21, dst 209.165.201.41).

```

```

Device#
*Feb 19 22:08:16.333: LISP: Send map request type SMR
*Feb 19 22:08:16.333: LISP: Send map request for EID prefix IID 0 172.16.0.12/32
*Feb 19 22:08:16.333: LISP-0: AF IID 0 IPv4, Send SMR triggered map request for 172.16.0.12/32
  (2) from 192.168.0.22.
*Feb 19 22:08:16.333: LISP-0: EID-AF IPv4, Sending map-request from 172.16.0.12 to 172.16.0.12
  for EID 172.16.0.12/32, ITR-RLOCs 1,
  nonce 0x4D04AB2F-0x99FF6FF5 (encap src 198.51.100.21, dst 209.165.201.41).
Device#
*Feb 19 22:08:18.423: LISP-0: Map Request IID 0 prefix 172.16.0.12/32 SMR[LL], Switching
  Map-Resolver 209.165.201.41 to 209.165.201.31.
*Feb 19 22:08:18.423: LISP: Send map request type SMR
*Feb 19 22:08:18.423: LISP: Send map request for EID prefix IID 0 172.16.0.12/32
*Feb 19 22:08:18.423: LISP-0: AF IID 0 IPv4, Send SMR triggered map request for 172.16.0.12/32
  (3) from 192.168.0.22.
*Feb 19 22:08:18.423: LISP-0: EID-AF IPv4, Sending map-request from 172.16.0.12 to 172.16.0.12
  for EID 172.16.0.12/32, ITR-RLOCs 1,
  nonce 0x5A4AC708-0x59A42AB6 (encap src 198.51.100.21, dst 209.165.201.31).
*Feb 19 22:08:18.424: LISP: Processing received Map-Reply(2) message on Ethernet0/0 from
  209.165.201.31:4342 to 198.51.100.21:4342
*Feb 19 22:08:18.424: LISP: Received map reply nonce 0x5A4AC708-0x59A42AB6, records 1

```

xTR2's map-cache is updated upon map-reply from the map server:

```

Device# show ip lisp map-cache
LISP IPv4 Mapping Cache for EID-table default (IID 0), 3 entries

0.0.0.0/0, uptime: 03:56:43, expires: never, via static send map-request
Negative cache entry, action: send-map-request
10.20.20.0/24, uptime: 00:24:04, expires: 23:35:56, via map-reply, complete
Locator      Uptime   State   Pri/Wgt
203.0.113.11 00:24:04 up      1/100
172.16.0.10/24, uptime: 00:59:48, expires: 00:00:51, via map-reply, forward-native
Negative cache entry, action: forward-native

```

xTR1 will put the 172.16.0.10/24 prefix in its away table:

```

Device# show ip lisp away
LISP Away Table for router lisp 0 (default) IID 0
Entries: 1
Prefix                               Producer
172.16.0.10/24                       local EID

```

Additional References for LISP Reliable Registration

Related Documents

Document Title	Location
Cisco IOS commands	Cisco IOS Master Command List, All Releases
LISP commands	Cisco IOS IP Routing: LISP Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 6830	<i>The Locator/ID Separation Protocol (LISP)</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for LISP Generalized SMR

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 9: Feature Information for LISP Generalized SMR

Feature Name	Releases	Feature Information
LISP Generalized SMR		<p>The LISP Generalize SMR feature supports LISP mobility, de-configured local Endpoint Identifier (EID), local EID no-route, overlapping prefix support, and mobility host move out and detection.</p> <p>The following commands were modified: show ip lisp away, show ip lisp data, show ip lisp map-cache.</p>



CHAPTER 15

TTL Propagate Disable and Site-ID Qualification

The TTL Propagate Disable feature supports disabling of the TTL (Time-To-Live) propagation for implementing the traceroute tool in a LISP network when RLOC and EID belong to different address-family.

The Site ID Qualification feature supports Endpoint Identifier (EID) prefix registration by multiple LISP sites.

- [Information About TTL Propagate Disable and Site-ID Qualification, on page 243](#)
- [How to Configure Site ID Qualification, on page 246](#)
- [How to Disable TTL Propagation, on page 247](#)
- [Additional References for TTL Propagate Disable and Site-ID Qualification, on page 249](#)
- [Feature Information for TTL Propagate Disable and Site-ID Qualification, on page 249](#)

Information About TTL Propagate Disable and Site-ID Qualification

LISP Site

LISP site is a set of routers in an edge network that are under a single technical administration. LISP routers in the edge network are the demarcation points to separate the edge network from the core network.

Map Server (MS)

An MS implements part of the distributed LISP mapping database by accepting registration requests from its client Egress Tunnel Routers (ETRs) and aggregating the successfully registered EID prefixes of ETRs.

Routing Locator (RLOC)

An RLOC is an IPv4 or IPv6 address of an Egress Tunnel Router (ETR).

Traceroute Tool

The traceroute tool is used to discover the routes that packets take when traveling to their destination.

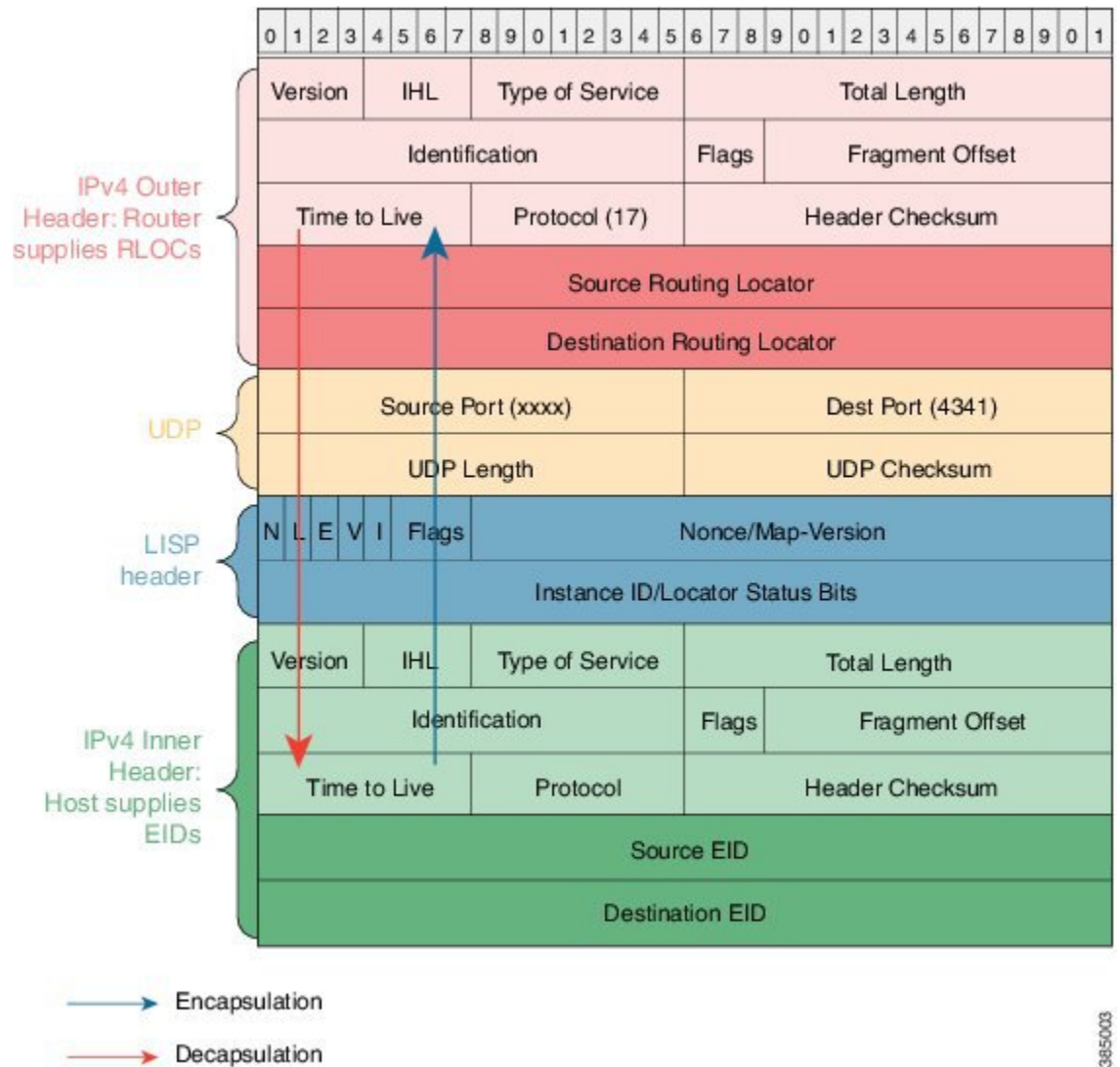
Site ID Qualification

A site is best conceptualized as an authentication domain: A set of ETRs under the same administrative control. The map server authenticates all ETRs in a site using the same shared key. Without the concept of a site, the map server would be required to have prior knowledge of every ETR in the network along with its authentication key. Site managers will not be able to deploy new ETRs without changing the configuration of the map servers. When a site is considered as an authentication domain as opposed to a topological grouping, then it is easy to see that the benefit of site ID qualification resides in the ability of reaching an EID prefix through ETRs under different administrative control.

With Site ID Qualification, the map server can have the same prefix configuration under multiple sites. The name of the feature stems from the requirement that any two sites with at least one prefix in common must be qualified with a unique site IDs.

TTL Propagation

Figure 50: TTL Propagation Mechanism



TTL Propagation mechanism as shown in the figure is described below:

- A LISP ITR encapsulates a packet and copies TTL value from inner header to outer header.
- A LISP ETR decapsulates a packet and copies TTL value from outer header to inner header if the outer header TTL value is smaller than the inner header TTL.

When TTL propagation is enabled the traceroute tool can display all middle hops between an LISP ITR and ETR. However, when RLOC and EID are of different address-family the traceroute output is undesirable.

When the above cross address-family situation exists, LISP does not propagate TTL between inner and outer IPv4 or IPv6 headers. During encapsulation, ITR uses the maximum permissible TTL in the outer header instead of using the TTL value from the inner header.

It is better to make the LISP tunnel between the ITR and ETR appear as a single hop to the client of traceroute. This is done through the `disable-ttl-propagate` configuration CLI either for a specific `eid-table` or the entire router lisp tag.



Note The TTL propagation is turned on automatically.

How to Configure Site ID Qualification

Configuring Site ID Qualification

```

site A
  conf t
  router lisp
  site A
  site-id 1
  authentication-key key1
  eid-prefix 1.2.0.0/16 accept-more-specifics

site B
  conf t
  router lisp
  site A
  site-id 1
  authentication-key key2
  eid-prefix 1.2.0.0/16 accept-more-specifics

```

Example: Site ID Qualification

When a site ID registration is received, the map server searches for the longest matching configured prefix. If the resulting prefix is less specific than the registration and does not have "accept-more-specifics" keyword, the registration is rejected; otherwise it is authenticated using the key of the site associated with the prefix. In this example "lazy" map server configuration is used so that an ETR can register any prefix with the map server.

Lazy Map Server Configuration:



Note Setup a new MSMR that has the same lazy configuration for two different sites.

```

enable
conf t
router lisp
locator-table default
site A
  site-id 100
  authentication-key key1
  eid-prefix 0.0.0.0/0 accept-more-specifics
  eid-prefix 2000:AAAA:BBBB::/96 accept-more-specifics
exit

```



```
!  
site B  
  site-id 200  
  authentication-key key2  
  eid-prefix 0.0.0.0/0 accept-more-specifics  
  eid-prefix 10.0.0.0/8 accept-more-specifics  
  eid-prefix 21.0.0.0/8 accept-more-specifics  
  eid-prefix 2000:BBBB:AAAA::/96 accept-more-specifics  
  exit  
  ipv4 map-server  
  ipv4 map-resolver  
  ipv6 map-server  
  ipv6 map-resolver  
  exit
```

How to Disable TTL Propagation



Note The TTL propagation can be disabled for a specific EID-table or an entire router LISP tag.

Disabling TTL Propagation for EID-Table

```
enable  
configure terminal  
router lisp  
  eid-table default instance-id 0  
  disable-ttl-propagate  
end
```

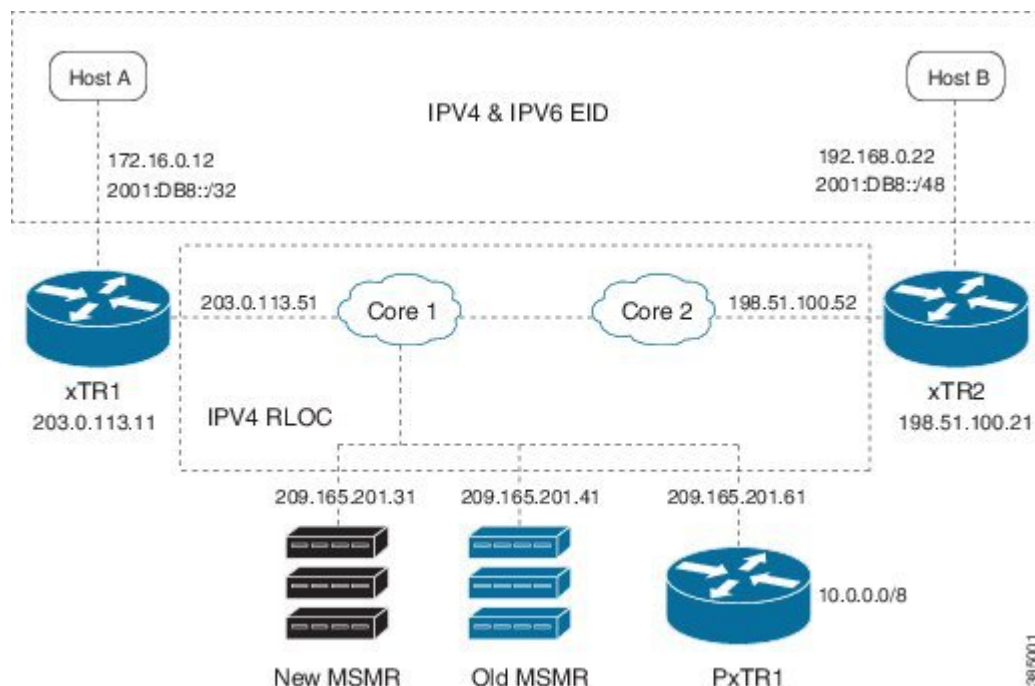
Disabling TTL Propagation for Router LISP Tag

```
enable  
configure terminal  
router lisp  
  disable-ttl-propagate  
end
```

Verifying TTL Propagate Disable

Perform this task to verify the TTL Propagate Disable feature which is enabled automatically in the LISP network. In this example, a LISP site uses a single edge router that functions as both ITR and ETR (known as an xTR). Routing Locators (RLOCs) are in IPv4. EID prefixes are in both IPv4 and IPv6. The LISP site registers to two map server/map resolver (MSMR) devices in the network core. The topology used in verifying TTL Propagate Disable is as shown in the figure below.

Figure 51: TTL Propagate Disable Topology



The components as shown in the topology are described below:

- xTR1 and xTR2 are xTRs for 2 LISP sites.
- Core1 and Core 2 are routing locators (RLOCs) core routers with no LISP configuration.
- New MSMR is a map-server and map-resolver with reliable map-registration support, whereas Old MSMR does not support reliable map-registration.
- PxTR1 works as a Proxy Ingress Tunnel Router (PITR) and Proxy Egress Tunnel Router (PETR) between the network with 10.0.0.0/8 prefix and the LISP sites.
- Only static routing protocols are used in this setup to reduce control traffic.



Note An IPv6 EID and IPv4 RLOC traceroute output will hide the middle hops between ITR and ETR even when TTL propagation is not disabled.

After disabling TTL propagation, an IPv4 EID over IPv4 RLOC traceroute output appears as below on Host A:

```
Device# traceroute 192.168.0.22

Type escape sequence to abort.
Tracing the route to 192.168.0.22
VRF info: (vrf in name/id, vrf out name/id)
 1 203.0.113.11 1 msec 1 msec 0 msec
 2 10.40.40.21 1 msec 1 msec 1 msec
 3 192.168.0.22 0 msec 2 msec *
```

Additional References for TTL Propagate Disable and Site-ID Qualification

Related Documents

Document Title	Location
Cisco IOS commands	Cisco IOS Master Command List, All Releases
LISP commands	Cisco IOS IP Routing: LISP Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 6830	<i>The Locator/ID Separation Protocol (LISP)</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for TTL Propagate Disable and Site-ID Qualification

Table 10: Feature Information for TTL Propagate Disable and Site-ID Qualification

Feature Name	Releases	Feature Information
TTL Propagate Disable and Site-ID Qualification		<p>The TTL Propagate Disable feature supports disabling of the TTL (Time-To-Live) propagation for implementing the traceroute tool in a LISP network when RLOC and EID belong to different address-family.</p> <p>The Site ID Qualification feature supports Endpoint Identifier (EID) prefix registration by multiple LISP sites.</p> <p>The following commands were modified: disable-ttl-propagate, eid-prefix, eid-table, router lisp, site-id, traceroute.</p>



CHAPTER 16

DNA SA Border Node Support

Digital Network Architecture (DNA) Security Access (SA) is an Enterprise architecture that brings together multiple building blocks needed for a programmable, secure, and highly automated fabric. Secure Fabric forms the foundation of this architecture and is targeted to address next generation campus trends. From Cisco IOS XE Everest 16.4.1 release, ASR 1000/ISR 4000 platforms can be supported as the border node of DNA SA fabric, handing off the enterprise campus fabric to iWAN, providing IP connectivity across campus and branches. The fabric is separated for campus and branches, and the border node will hand off the LISP/VxLAN-GPO fabric to WAN. In the 16.4.1 release, the handoff is to the DMVPN/MPLS WAN with manual configuration.

- [Finding Feature Information, on page 251](#)
- [Restrictions for DNA SA Border Node Support, on page 251](#)
- [Information About DNA SA Border Node Support, on page 252](#)
- [Configuration Example: Border Node as LISP PxTR, on page 255](#)
- [Configuration Example: Border Node as LISP xTR, on page 259](#)
- [Feature Information for DNA SA Border Node Support, on page 261](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for DNA SA Border Node Support

- IPv6 RLOC and IPv6 EID is not supported for DNA SA.
- IPv4 SGT can control (enable or disable) IPv4/IPv6 EID SGT. IPv6 SGT is not supported.
- Multicast configuration cannot change encapsulation type.

Information About DNA SA Border Node Support

Enabling VxLAN Encapsulation for LISP Control Plane

To enable VXLAN encapsulation for LISP, use the encapsulation vxlan command in the router lisp configuration mode. This command must be configured on all LISP edge devices in the enterprise fabric deployment: Ingress Tunnel Router (ITR), Egress Tunnel Router (ETR), Proxy Ingress Tunnel Router (PITR), Proxy Egress Tunnel Router (PETR). Failure to configure this command on any of the LISP edge devices will result in loss of control and data traffic.

Use the **show platform software lisp udp-src-port ipv4 src_ip dest_ip protocol** command to see the UDP source port according to the data packets. You can also use **ipv6** in the command.



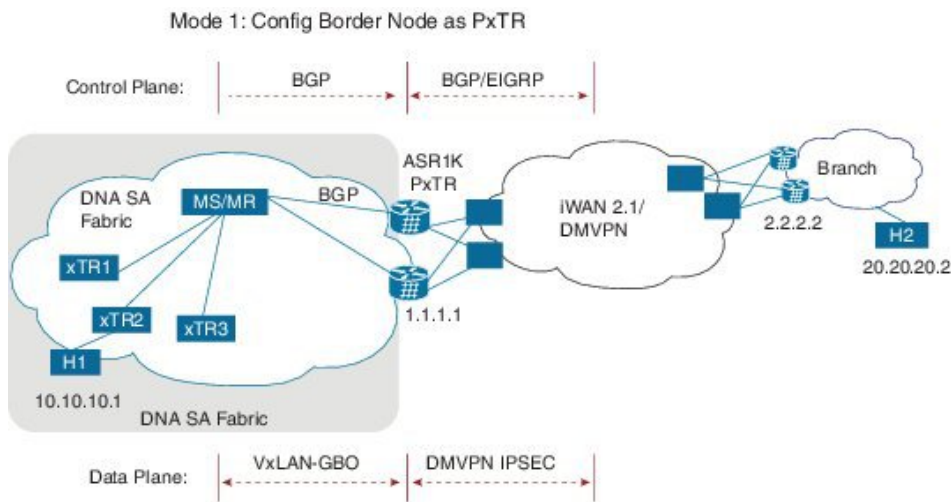
Note VXLAN must not be configuration on the device when VXLAN encapsulation is enabled for LISP. Conversely, VXLAN encapsulation for LISP must not be enabled when configuring other VXLAN protocols.

Two deployment modes are supported, one is to configure border node as PxTR and the other is to configure border node as XTR.

Configuring Border Node as LISP PxTR

Border node can be configured as PxTR for the fabric.

Figure 52: Border Node as LISP PxTR



380372

Control Plane Connectivity

Campus-to-Branches direction:

- xTR will register its direct attached host to MS/MR through LISP map-register.
- There will be per-VRF BGP sessions between MS/MR and PxTR, MS/MR will advertise LISP routes to PxTR
- PxTR will re-originate those routes to WAN through EIGRP or BGP.

Branches-to-Campus direction:

- Branch routes will advertise its routes to border nodes of campus through EIGRP or BGP.
- Border nodes (PxTR) will not advertise routes to LISP MS/MR.
- On XTR, configure “ipv4 use-petr <rloc of PxTR> ”

Packet Flow with Control Plan Interworking

H1 to H2: SIP:10.10.10.1, DIP: 20.20.20.2

- Assuming xTR2 is the default gateway for H1 (it might not be the access switch, but the distribution switch instead). H1 sends the IP packet to xTR2 after it resolves the ARP entry for gateway MAC.
- On xTR2, the IPv4 use-petr 2.2.2.2 is configured.
- On xTR2, a MAP request is initiated to MAP request, to resolve 20.20.20.2
- A negative MAP reply is sent from MS/MR to xTR2.
- xTR2 encapsulation with LISP head and sends to LISP PxTR 1.1.1.1
- Branch router 2.2.2.2 advertises 20.20.20/24 routes to border node 1.1.1.1 using WAN protocol BGP/EIGRP.
- PxTR send the packet to remote branch router 2.2.2.2 through iWAN/DMVPN.

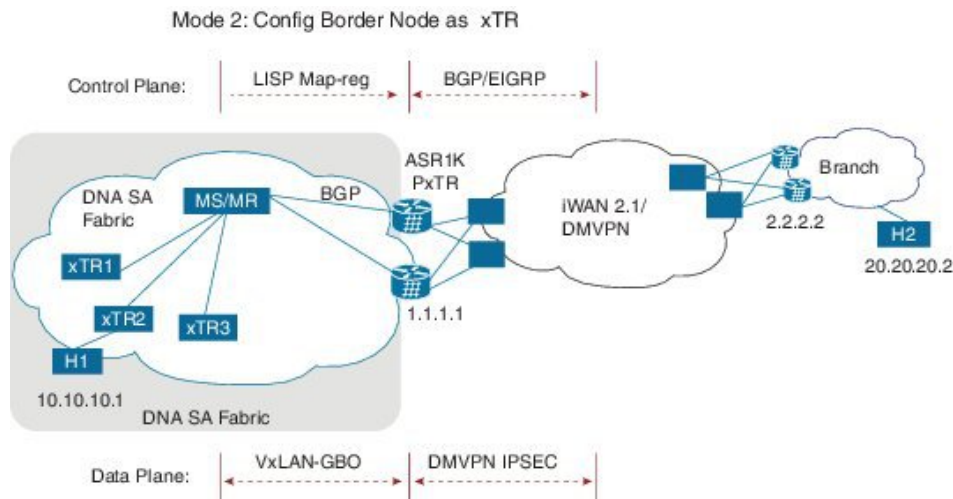
H2 to H1: SIP: 20.20.20.2, DIP: 10.10.10.1

- xTR2 register 10.10.10.1 to MS/MS through LISP MAP-register.
- MS/MR advertise this route to PxTR 1.1.1.1
- PxTR re-originate route to branch route 2.2.2.2
- H2 sends the packets to branch router 2.2.2.2
- Branch router 2.2.2.2 forwards the packets to PxTR 1.1.1.1
- PxTR sends MAP-request to resolve 10.10.10.1, and the MAP-reply is from xTR2.
- PxTR sends LISP packets to xTR2 and then to H1.

Configuring Border Node as LISP xTR

Border node can be configured as xTR for the fabric.

Figure 53: Border Node as LISP xTR



38/5373

Control Plane Connectivity

Campus-to-Branche direction--For each subnet of fabric, you must manually configure a static route to null0 on ASR1K xTR. Example: `ip route vrf vrf1 10.10.1 255.255.255.0 Null0 tag 110 ASR1K xTR (1.1.1.1)` will advertise this static route to remote branches (2.2.2.2) through BGP or EIGRP.

Branche-to-Campus direction--Remote Branche (2.2.2.2) will advertise routes 20.20.20.2 to ASR1K xTR (1.1.1.1) through BGP or EIGRP. On ASR1K xTR, configure “`ipv4 route-import database bgp 100 ...`” under LISP EID table to import BGP/EIGRP as LISP EID table. ASR1K xTR 2.2.2.2 will initiate MAP-register to register the EID learnt from BGP.

Packet Flow with Control Plan Interworking

H1 to H2: SIP: 10.10.10.1, DIP: 20.20.20.2

- Branche route 2.2.2.2 advertises routes 20.20.20.0/24 to LISP xTR 1.1.1.1 through BGP/EIGRP.
- LISP xTR 1.1.1.1 will import 20.20.20.0/24 into local EID table.
- LISP xTR 1.1.1.1 sends MAP-register to MS/MR to register 20.20.20.0/24 as its local EID
- H1 sends IP packets to xTR2 after it resolves the MAC address of xTR2.
- xTR2 sends map-request to resolve the device for 20.20.20.2 and the RLOC is 1.1.1.1
- xTR2 sends VxLAN encapsulated packets to 1.1.1.1
- RLOC 1.1.1.1 terminates VxLAN and forwards the packets to 2.2.2.2.

H2 to H1: SIP: 20.20.20.2, DIP: 10.10.10.1

- Static route of 10.10.10.1/24 is configured on xTR 1.1.1.1 and it points to null0
- xTR advertises this route to branche 2.2.2.2
- H2 sends packets to branche router 2.2.2.2
- Branche router forwards the packets to LISP xTR 1.1.1.1
- Branche router 2.2.2.2 forwards the packets to PxTR 1.1.1.1
- On LISP xTR 1.1.1.1, 10.10.10.1/24 is pointed to null0, which will trigger LISP routing; it will send MAP-request to resolve the RLOC for 10.10.10.1.
- LISP xTR 1.1.1.1 sends VxLAN encapsulated packets to xTR2.

Security Group Tag (SGT) Propagation

Besides the control plane and data plane connectivity, the SGT tag must be carried over from the campus fabric to WAN and vice-versa, so that SGT tag based policy will be enforced end-to-end across campus and branches. This function has dependence on WAN; if the WAN cannot carry the SGT tag, the tag will be lost.

Configuration Example: Border Node as LISP PxTR

Border node configuration:

```
vrf definition vrf1
  rd 1:1
  !
  address-family ipv4
    route-target export 1:1
    route-target import 1:1
  exit-address-family
  !
vrf definition vrf2
  rd 1:2
  !
address-family ipv4
  route-target export 1:2
  route-target import 1:2
exit-address-family
!
interface Loopback1
  vrf forwarding vrf1
  ip address 7.7.7.7 255.255.255.255
  !
interface Tunnel100
  description "iwan tunnel for vrf1"
  vrf forwarding vrf1
  ip address 100.0.0.1 255.255.255.0
  tunnel source GigabitEthernet2
  tunnel destination 16.0.0.2
  tunnel key 100
  !
interface Tunnel101
  description "iwan tunnel for vrf2"
  vrf forwarding vrf2
  ip address 101.0.0.1 255.255.255.0
  tunnel source GigabitEthernet2
  tunnel destination 16.0.0.2
  tunnel key 101
  !
interface Tunnel1000
  description "pxtr and msr tunnel vrf1"
  vrf forwarding vrf1
  ip address 200.0.0.2 255.255.255.0
  tunnel source GigabitEthernet1
  tunnel destination 13.0.0.1
  tunnel key 1000
  !
interface Tunnel1001
  description "pxtr and msr tunnel vrf2"
  vrf forwarding vrf2
  ip address 201.0.0.2 255.255.255.0
  tunnel source GigabitEthernet1
```

```

tunnel destination 13.0.0.1
tunnel key 1001
!
interface GigabitEthernet1
 ip address 15.0.0.2 255.255.255.0
 ip ospf 1 area 0
!
interface GigabitEthernet2
 ip address 16.0.0.1 255.255.255.0
!
router lisp
 encapsulation vxlan //Enable VXLAN GPO encapsulation for the LISP data plane//
 eid-table default instance-id 0
 map-cache 0.0.0.0/0 map-request
 exit
!
eid-table vrf vrf1 instance-id 1
 ipv4 route-import map-cache bgp 100 route-map set_lisp_vrf1
 exit
!
eid-table vrf vrf2 instance-id 2
 ipv4 route-import map-cache bgp 100 route-map set_lisp_vrf2
 exit
!
ipv4 sgt //enable SGT function for SGT tag propagation//
 exit
!
ipv4 map-request-source 14.0.0.2
ipv4 proxy-etr
ipv4 proxy-itr 15.0.0.2
ipv4 itr map-resolver 14.0.0.1
 exit
!
router ospf 1
!
router bgp 100
 bgp log-neighbor-changes
!
 address-family ipv4 vrf vrf1
  neighbor 100.0.0.2 remote-as 100
  neighbor 100.0.0.2 activate
  neighbor 200.0.0.1 remote-as 200
  neighbor 200.0.0.1 ebgp-multihop 255
  neighbor 200.0.0.1 update-source Tunnel1000
  neighbor 200.0.0.1 activate
  neighbor 200.0.0.1 send-community both
 exit-address-family
!
 address-family ipv4 vrf vrf2
  neighbor 101.0.0.2 remote-as 100
  neighbor 101.0.0.2 activate
  neighbor 201.0.0.1 remote-as 200
  neighbor 201.0.0.1 ebgp-multihop 255
  neighbor 201.0.0.1 update-source Tunnel1001
  neighbor 201.0.0.1 activate
  neighbor 201.0.0.1 send-community both
 exit-address-family
!
 ip bgp-community new-format
 ip community-list 10 permit 1000:1
 ip community-list 11 permit 1000:2
!
 route-map set_lisp_vrf1 permit 10
 match community 10

```

```

!
route-map set_lisp_vrf2 permit 10
match community 11
!
!
MSMR configuration:
vrf definition vrf1
 rd 1:1
 !
 address-family ipv4
 exit-address-family
 !
vrf definition vrf1000
 rd 1000:1
 !
 address-family ipv4
 exit-address-family
 !
vrf definition vrf2
 rd 1:2
 !
 address-family ipv4
 exit-address-family
 !
interface Loopback0
 ip address 14.0.0.1 255.255.255.255
 ip ospf 1 area 0
 !
interface Tunnel1000
 description "pxtr and msmr tunnel vrf1"
 vrf forwarding vrf1
 ip address 200.0.0.1 255.255.255.0
 tunnel source GigabitEthernet3.6
 tunnel destination 15.0.0.2
 tunnel key 1000
 !
interface Tunnel1001
 description "pxtr and msmr tunnel vrf2"
 vrf forwarding vrf2
 ip address 201.0.0.1 255.255.255.0
 tunnel source GigabitEthernet3.6
 tunnel destination 15.0.0.2
 tunnel key 1001
 !
interface GigabitEthernet2
 no ip address
 !
interface GigabitEthernet2.4
 encapsulation dot1Q 4
 ip address 12.0.0.2 255.255.255.0
 ip ospf 1 area 0
 !
interface GigabitEthernet2.5
 encapsulation dot1Q 5
 ip address 12.0.1.2 255.255.255.0
 ip ospf 1 area 0
 !
interface GigabitEthernet3
 no ip address
 negotiation auto
 cdp enable
 !
interface GigabitEthernet3.6
 encapsulation dot1Q 6

```

```

ip address 13.0.0.1 255.255.255.0
ip ospf 1 area 0
!
interface GigabitEthernet3.7
 encapsulation dot1Q 7
 ip address 13.0.1.1 255.255.255.0
 ip ospf 1 area 0
!
router lisp
 eid-table default instance-id 0
  exit
!
eid-table vrf vrf1 instance-id 1
 ipv4 route-export site-registrations
  exit
!
eid-table vrf vrf2 instance-id 2
 ipv4 route-export site-registrations
  exit
!
rtr-set rtr
 12.0.0.1 authentication-key cisco
 12.0.1.1 authentication-key cisco
  exit
!
map-server advertise-rtr-set rtr
site xtr1
 authentication-key cisco
 advertise-rtr-set rtr
 eid-prefix 1.1.1.1/32 route-tag 110
 eid-prefix instance-id 1 5.5.5.5/32 route-tag 100
  exit
!
site xtr2
 authentication-key cisco
 eid-prefix 2.2.2.2/32 route-tag 110
 eid-prefix instance-id 1 6.6.6.6/32 route-tag 100
 eid-prefix instance-id 1 11.11.11.11/32 route-tag 120
 eid-prefix instance-id 2 6.6.6.6/32 route-tag 110
  exit
!
ipv4 map-server
ipv4 map-resolver
exit
!
router ospf 1
!
router bgp 200
  bgp log-neighbor-changes
!
address-family ipv4 vrf vrf1
 redistribute lisp metric 11 route-map set_lisp_vrf1
 neighbor 200.0.0.2 remote-as 100
 neighbor 200.0.0.2 ebgp-multihop 255
 neighbor 200.0.0.2 update-source Tunnel1000
 neighbor 200.0.0.2 activate
 neighbor 200.0.0.2 send-community both
 exit-address-family
!
address-family ipv4 vrf vrf2
 redistribute lisp metric 11 route-map set_lisp_vrf2
 neighbor 201.0.0.2 remote-as 100
 neighbor 201.0.0.2 ebgp-multihop 255
 neighbor 201.0.0.2 update-source Tunnel1001

```

```

neighbor 201.0.0.2 activate
neighbor 201.0.0.2 send-community both
exit-address-family
!
!
ip bgp-community new-format
!
route-map set_lisp_vrf1 permit 10
match tag 100
set community 1000:1
!
route-map set_lisp_vrf2 permit 10
match tag 110
set community 1000:2
!

```

Configuration Example: Border Node as LISP xTR

Border node configuration:

```

vrf definition vrf1
rd 1:1
!
address-family ipv4
route-target export 1:1
route-target import 1:1
exit-address-family
!
vrf definition vrf2
rd 1:2
!
address-family ipv4
exit-address-family
!
interface Loopback0
ip address 2.2.2.2 255.255.255.255
!
interface Loopback1
vrf forwarding vrf1
ip address 6.6.6.6 255.255.255.255
!
interface Tunnel200
description "iWAN tunnel to remote branch"
vrf forwarding vrf1
ip address 150.0.0.2 255.255.255.0
tunnel source GigabitEthernet2
tunnel destination 17.0.0.1
tunnel key 200
!
interface GigabitEthernet2
ip address 17.0.0.2 255.255.255.0
!
interface GigabitEthernet3
no ip address
!
interface GigabitEthernet3.6
encapsulation dot1Q 6
ip address 13.0.0.2 255.255.255.0
ip ospf 1 area 0
!
interface GigabitEthernet3.7

```

```

encapsulation dot1Q 7
ip address 13.0.1.2 255.255.255.0
ip ospf 1 area 0
!
interface GigabitEthernet4
ip address 15.0.0.1 255.255.255.0
ip ospf 1 area 0
!
router lisp
encapsulation vxlan
locator-set set1
13.0.0.2 priority 1 weight 1
13.0.1.2 priority 1 weight 1
exit
!
eid-table default instance-id 0
database-mapping 2.2.2.2/32 locator-set set1
exit
!
eid-table vrf vrf1 instance-id 1
database-mapping 6.6.6.6/32 locator-set set1
ipv4 route-import database bgp 100 route-map match_com locator-set set1
exit
!
eid-table vrf vrf2 instance-id 2
database-mapping 6.6.6.6/32 locator-set set1
exit
!
ipv4 sgt //enable SGT function for SGT tag propagation//
exit
!
ipv4 use-petr 15.0.0.2
ipv4 itr map-resolver 14.0.0.1
ipv4 itr
ipv4 etr map-server 14.0.0.1 key cisco
ipv4 etr
exit
!
router ospf 1
!
router bgp 100
bgp log-neighbor-changes
!
address-family ipv4 vrf vrf1
redistribute static route-map tag_110
neighbor 150.0.0.1 remote-as 100
neighbor 150.0.0.1 activate
neighbor 150.0.0.1 send-community both
exit-address-family

ip bgp-community new-format
ip community-list 10 permit 200:1
ip route vrf vrf1 5.5.5.5 255.255.255.255 Null0 tag 110
!
route-map tag_110 permit 10
match tag 110
!
route-map match_com permit 10
match community 10
!

```

Feature Information for DNA SA Border Node Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 11: Feature Information for DNA SA Border Node Support

Feature Name	Releases	Feature Information
DNA SA Border Node Support	Cisco IOS XE Everest 16.4.1 Release	From Cisco IOS XE Everest 16.4.1 release, ASR 1000/ISR 4000 platforms can be supported as the border node of DNA SA fabric, handing off the enterprise campus fabric to iWAN, providing IP connectivity across campus and branches.

