データプレーンセキュリティの実装

データプレーンセキュリティ（DPSec）機能は、外部送信元から LISP VPN へのトラフィック注入を防止します。DPSec は、Unicast Reverse Path Forwarding（uRPF）のサポートを使用して構築されたルーティングロケータ（RLOC）ネットワークの整合性に依存します。

認証や暗号化のオーバーヘッドを発生させずに LISP 共有モードセグメンテーションを有効にするため、DPSec 機能では、ネットワーク上で URPF を適用する送信元 RLOC カプセル化解除フィルタリングというメカニズムを使用します。ネットワークに設定されている URPF は、URPF によってすでに証明されているトラフィックの許容可能 RLOC のリストを配布します。これにより、LISP 制御およびデータパケットの送信元 RLOC アドレスをスプーフィングできなくなります。DPSec 機能では、各 EID インスタンスの有効なカプセル化送信元のリストを使用して、xTR と PxTR でのカプセル化解除時に LISP データパケットのフィルタリングを行います。

（注）

- LISP 転送は Cisco ASR 9000 高密度100GE イーサネットラインカードでサポートされていますが、LISP IPv6 RLOC および LISP データプレーンセキュリティ機能は、これらのカードではサポートされていません。

データプレーンセキュリティの機能の履歴

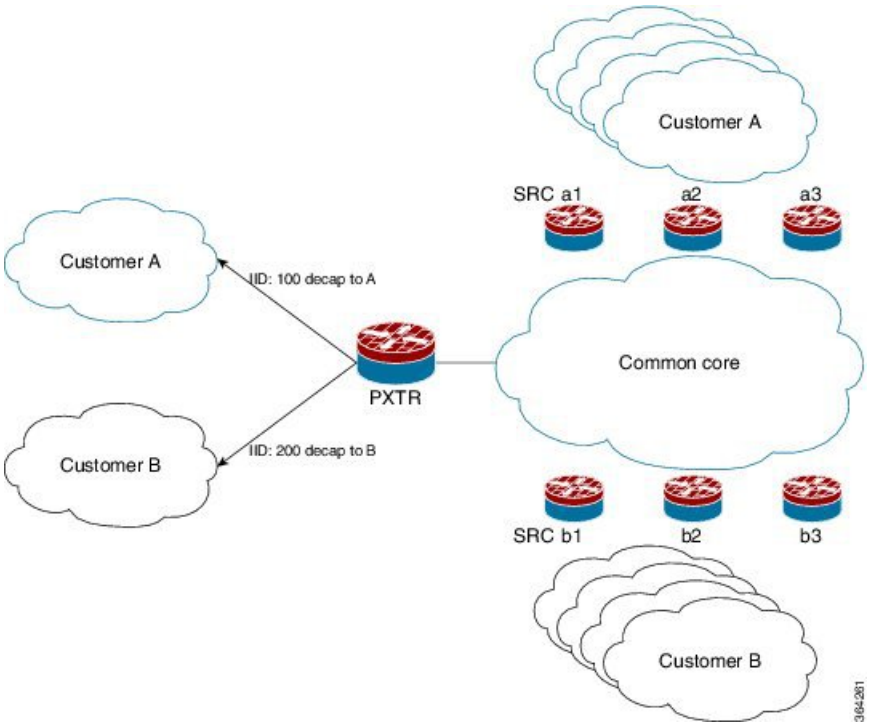
リリース 5.3.0	この機能が導入されました。

データプレーンセキュリティに関する情報

LISP データ プレーン セキュリティ機能により、LISP VPN からのトラフィックのみが VPN でカプセル化を解除できます。データプレーンセキュリティを理解するには、それがサポートしている次の機能と概念を十分に理解しておく必要があります。

送信元 RLOC カプセル化解除のフィルタリング

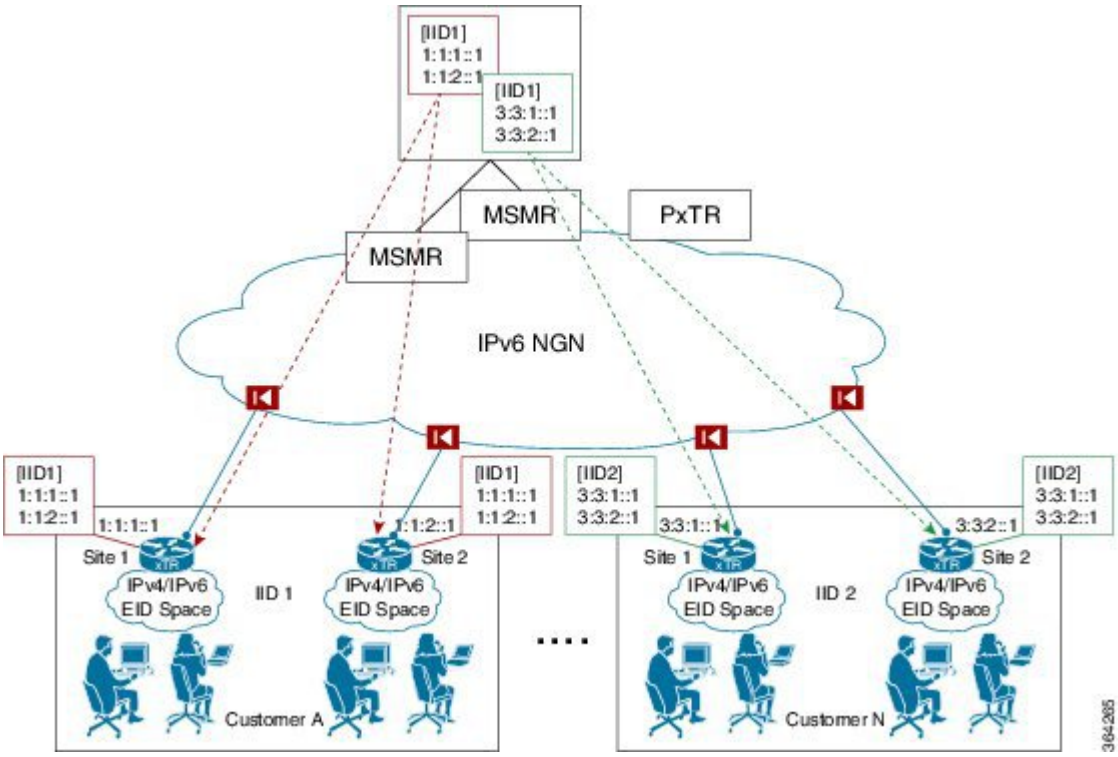
次に、共有共通 RLOC コアを介して、LISP EID インスタンス ID（IID）100 および 200 を使用しているカスタマーネットワークをそれぞれ青と黒で示した図を示します。LISP データパケットをカプセル化を解除すると、PxTR は、インスタンス ID 100 を伝送しているパケットの a1、a2、または a3 のカプセル化ヘッダー内に送信元（SRC）RLOC があることを検証します。同様に、インスタンス ID 200 の場合、PxTR は RLOC 送信元が b1、b2、または b3 であることを検証します。



有効な送信元 RLOC を伝送しない LISP カプセル化データパケットはドロップされます。RLOC 空間 URPF の適用と送信元 RLOC ベースのカプセル化解除フィルタリングを組み合わせることで、テナント VPN のメンバーではない送信元が VPN にトラフィックを挿入できないようにします。

EID インスタンスメンバーシップの配布

送信元 RLOC フィルタリング ソリューションを展開するには、マッピングシステムを介して有効な RLOC のリストをカプセル化解除を実行するボックスにプッシュするための自動化されたメカニズムが必要です。この機能は、マップサーバによって実行されます。マップサーバは、Map-Register メッセージで受信したマッピングレコード内の RLOC 情報を使用して、EID インスタンス ID と RLOC のメンバーシップリストを作成します。EID インスタンス ID で識別された VPN のパケットのカプセル化解除が必要なすべての xTR と PxTR に完全なリストがプッシュされます。



この例では、マップサーバがカスタマーごとに個別のVPN（EIDインスタンス）メンバーシップリストを作成し、リストの内容をプッシュします。カスタマー A の 2 つの xTR はそれぞれのサイトの RLOC を登録します。各ユーザは、マップサーバから、カスタマー A のすべての xTF の RLOC の完全なリストを受信します。受信したリストは、カプセル化解除トラフィックをフィルタリングし、データプレーンのセキュリティを適用するために使用されます。

PxTR が使用されている場合（VPN へのインターネット接続をVPN に提供するなど）、VPN に参加している xTR は、PxTR によって送信された LISP データパケットを受け入れ、カプセル化を解除する必要があります。PxTR によって使用される RLOC アドレスは、マップサーバによって xTR に伝達される EID インスタンス メンバーシップ リストに含まれている必要があります。PxTR は、マップサーバが PxTR RLOC を検出するために使用できるマップサーバに EID プレフィックスを登録しません。これらの RLOC はマップサーバ上に手動で設定する必要があります。

マップサーバによって構築された EID インスタンスのメンバーシップリストが有効なのは、VPN に参加しているボックスのみです。追加されたセキュリティ対策として、マップサーバは、EID インスタンスのメンバーシップリストの内容を、その VPN のメンバである xTR と PxTR にのみ伝達します。

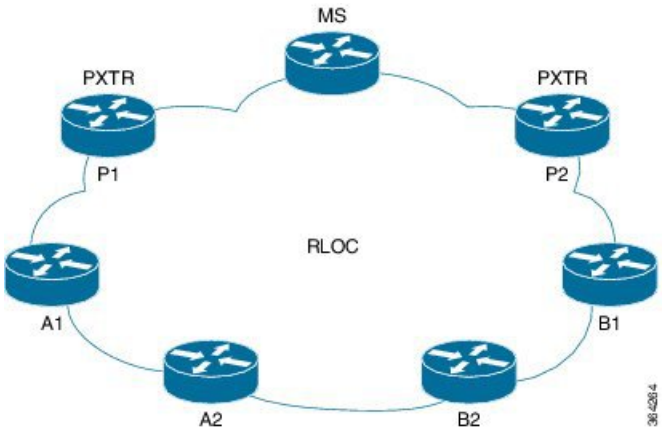
マップサーバメンバーシップの収集と配布

LISP マップサーバは、EID インスタンス単位のメンバーシップを追跡し、それを (P)xTR に配布する役割を担います。この機能を有効にするには、**map-server rloc members distribute** コマンドを使用します。このコマンドは、マップサーバを次のように設定します。

- 信頼性の高いトランスポートセッションを受け入れるためのマップ登録と設定を使用して、RLOC アドレスのリストを構築する。
- 上記のリストの (P)xTR からの TCP 接続を受け入れる。
- 受信した Map-Register メッセージから EID インスタンス単位で RLOC メンバーシップを収集し、保持する。
- (P)xTR からの信頼性の高い転送セッションを介して受信した EID インスタンスのメンバーシップ要求に対応し、メンバーシップ情報を配信する。

受信した登録から MS が収集した EID インスタンス単位のメンバーシップリストは、**map-server rloc members {add | override}** コンフィギュレーション コマンドを使用して拡張したり、完全に上書きすることができます。このコマンドを使用すると、検出した xTR RLOC メンバーシップを PxTR RLOC アドレスを使用して拡張できます。拡張されたメンバーシップリストは、信頼性の高い転送セッションを介して受信したメンバーシップ要求を許可するかどうかを決定するために使用されます。EID インスタンスに登録されている xTR からの要求のみが許可されます。拡張されたメンバーシップリストは、カプセル化解除デバイスにプッシュされてデータプレーンセキュリティ機能が実装された後、有効な xTR と PxTR の両方から送信されたカプセル化されたパケットを受け入れられるようになります。

マップサーバとの TCP 接続を確立しようとする無許可の試行を防ぐため、接続を許可する許可済みのロケータのリストが構築されます。このリストには、登録する xTR の RLOC アドレスとともに、メンバーシップリストの拡張で設定された RLOC のアドレスが含まれています。RLOC アドレスファミリごとに接続を受け入れる単一のリストがあることに注意してください（EID インスタンス固有ではありません）。



たとえば、上の図の 2 つの VPN があるネットワークを考えてみましょう。VPN A と B にはそれぞれ 2 つの xTR A1/A2 と B1/B2 があります。VPN A のメンバーシップは、「map-server rloc members add …」設定を使用して PxTR RLOC アドレス P1 を含めるように MS 上で拡張されます。VPN B のメンバーシップは、PxTR RLOC アドレス P2 を含むように拡張されます。MS が管理する結果のリストは次のようになります。

- EID インスタンス 1（VPN A）メンバーシップ：A1、A2、P1
- EID インスタンス 2（VPN B）メンバーシップ：B1、B2、P2

- TCP セッションを受け入れるロケーター：A1、A2、P1、B1、B2、P2

マップサーバは、確立された信頼性の高いトランスポートセッションごとに 1 つ以上の EID インスタンスの EID インスタンスメンバーシップ要求を受信する場合があります。PxTR は通常、MS で確立された 1 つのセッションを通じて複数のインスタンスのメンバーシップを要求します。マップサーバは、許可された要求ごとに完全なメンバーシップのリフレッシュと増分更新を提供する必要があります。

メンバーシップ要求が MS によって受信され、要求を発信しているピア (P)xTR が要求に関連する EID インスタンスのメンバではない場合、MS は要求を拒否し、(P)xTR に Membership-NACK メッセージを返します。このようなイベントは、通常の動作中に発生することがあります。これは、TCP セッションと xTR からのメンバーシップ要求を、対応する Map Register メッセージの前に受信して EID インスタンスメンバーシップに配置することがあるためです。EID インスタンスメンバーシップ要求が MS によって受け入れられた後、登録の有効期限または設定の変更が原因で、要求側 (P)xTR が EID インスタンスメンバーシップから削除された場合、MS は (P) xTR にそのインスタンスのメンバーシップ更新を受信しなくなったことを示す Membership-NACK メッセージを送信します。

マップサーバが再起動すると、メンバーシップ要求に対応する前に、まず、EID インスタンスのメンバーシップリストを検出して再構築する必要があります。特に、完全なメンバーシップリストを持たない EID インスタンスのメンバーシップリフレッシュ終了メッセージの送信は、MS でオフにする必要があります。MS では、LISP コントロールプレーンはメンバーシップリストの完了を検討する前に、登録の受信を待機します。次の条件を満たす必要があります。

- 最初の登録が受信されてから、次のいずれかの条件が満たされた後、少なくとも 1 つの登録期間が経過した（1 分）。
  - accept-more-specific site という EID プレフィックス設定が EID インスタンスに存在せず、設定済みのすべての EID プレフィックスの登録が受信されている。
  - 最初の登録が受信された時点から登録の 3 期間が経過しています。
  - 登録が受信されておらず、LISP コントロールプレーンが再起動してから 3 回の登録期間が経過しました。

EXEC コンフィギュレーション モードで **show lisp site rloc members** コマンドを使用して、メンバーシップの配布をマップサーバで管理できます。

データプレーンセキュリティの実装方法 （7 ページ） に、手順を詳細に示します。

(P)xTR でのカプセル化解除のフィルタリング

送信元 RLOC のカプセル化解除 RLOC フィルタリング機能は、**decapsulation filter rloc source** コマンドを使用して、(P)xTR 上で有効になります。この機能を有効にすると、(P)xTR はフィルタによって許可された送信元 RLOC を伝送する LISP データパケットのカプセル化解除のみを許可します。この機能を初めて有効にしたときに、フィルタがマップサーバからの EID インスタンスメンバーシップの自動検出に基づいている場合、マップサーバとの信頼性の高い転送接続が確立され、メンバーシップを受信するまで、トラフィックはドロップされます。

(P)xTR メンバーシップの検出

decapsulation filter rloc source members 設定を使用して 1 つ以上の EID インスタンスのメンバーシップ自動検出を使用したデータプレーンの送信元RLOCフィルタリング用に設定されている (P)xTR は、それらのインスタンスに設定されているマップサーバそれぞれとの信頼性の高い転送セッションを確立しようとします。1 つ以上の EID インスタンスのメンバーシップを伝達する各マップサーバを使用して、信頼性の高い1つの転送セッションが開始されます。自動検出されたメンバーシップリストは、**decapsulation filter rloc source** コマンドの **locator-set** オプションを使用して送信元 RLOC フィルタを形成するように拡張されています。各マップサーバから検出されたEIDインスタンスのメンバーシップリストは、設定されているロケータセットの内容とともにまとめてマージされ、データプレーンの送信元RLOCを定義するために使用されます。マップサーバは、最初に正常に登録した EID プレフィックスを持つ RLOC アドレスからの信頼性の高い着信転送接続のみを受け入れます。xTRは、正常に登録されたことを確認する Map-Notify を受信した後にのみ、接続を確立しようとします。特定のインスタンス ID の EID インスタンスメンバーシップを要求するには、そのインスタンスの 1 つ以上の EID プレフィックスが正常に登録されている必要があります。

マップサーバとの接続が確立されると、(P)xTR は設定内にマップサーバがある各 EID インスタンスの Membership-Request メッセージを送信します。受信した Membership-Add メッセージと Membership-Delete メッセージは、(P)xTR 上の EIDインスタンス メンバーシップ データベースを更新します。

EID インスタンス メンバーシップ データベースを再構築するために、(P)xTR は、Membership-ACK メッセージを使用してメンバーシップサービスの提供を希望することをマップサーバが示すとすぐに、Membership-Refresh-Request メッセージを発行します。(P)xTR は、検出された各メンバーシップエントリのエポックを保持します。マップサーバから Membership-Refresh-Start メッセージを受信すると、(P)xTR は、マップサーバと EID インスタンスの組み合わせについて保持するエポックを増分させ、既存のメンバーシップ状態を失効としてフラグ付けします。リフレッシュ時に受信した後続の Membership-Add メッセージは、対応するエントリのエポックを更新します。Membership-Refresh-End メッセージを受信すると、(P)xTRは、リフレッシュ時に更新されていない古いエポックを伝送している、マップサーバから受信した EID インスタンスのメンバーシップエントリを削除します。

転送する通信のフィルタリング

LISP コントロールプレーンは、RIB を通じて情報を伝達するための RIB の Opaque ファシリティをテーブル配布の一部として、すべてのFIBインスタンスまで使用します。メッセージは次のように定義されます。

- RLOC AF および EID インスタンスの粒度ごとにフィルタの有効化状態を伝達する
- RLOC フィルタのエントリを伝達する

TCP ベースの信頼性の高いトランスポートセッション

LISP は、EID インスタンスのメンバーシップの配布に xTR とマップサーバ間の TCP ベースのセッションを使用します。信頼性の高いトランスポートセッションは、アクティブセッションまたはパッシブセッションの確立を（TCP ポート 4342 を使用）をサポートしています。この

場合、xTRがアクティブロール、マップサーバがパッシブロールを担います。セッションは、送信元 RLOC フィルタリングに基づいて、マップサーバ側からの有効な RLOC からのみ受け入れられます。サポートできる同時 TCP 接続の数は、OS ごとおよびプラットフォームごとに異なります。次に、考慮する必要があるセキュリティ上の事項を示します。

- マップサーバが対応できる xTR の数は、プラットフォームで確立および保持できる TCP セッションの数で制限されます。これにより、マップサーバがホストできる VPN カスタマーの数が決定します。水平スケーリングは、VPN カスタマーを複数の Map Server 間で分割することによって実現されます。
- 同じ VPN に属しているすべての xTR は同じマップサーバに登録する必要があります。マップサーバ TCP セッションのスケール制限よりも多くの xTR を持つ VPN は使用できません。
- 最初の成果物のセッション認証は、RLOC ネットワークの整合性に依存しており、パケットの送信元アドレスを使用して TCP セッションのみをフィルタリングします。

セッションの確立、信頼性の高いトランスポートメッセージ形式、キープアライブメッセージ、エラー通知メッセージなどの TCP ベースの信頼性の高いトランスポートセッションの詳細については、http:// tools.ietf.org /id/draft-kouvelas-lisp-reliable-transport-00.txt を参照してください。

データプレーンセキュリティの実装方法

ここでは、次の手順について説明します。

送信元 RLOC ベースのカプセル化解除のフィルタリングの有効化

LISP パケットのカプセル化を解除するときに送信元検証用のカプセル化解除フィルタリストをダウンロードするように xTR またはプロキシ xTR を設定するには、lisp コンフィギュレーション モードで **decapsulation filter source** コマンドを使用します。

(P)ETR が LISP パケットのカプセル化を解除する場合は、LISP パケットの外部ヘッダー送信元アドレスを考慮せずに実行されます。送信元アドレスが信頼できるネットワーク環境では、カプセル化解除前に LISP パケットの送信元アドレスを考慮する必要がある場合があります。(P)xTR で decapsulation filter source コマンドを設定すると、デバイスはマップサーバとの信頼性の高い TCP ベースのトランスポートセッションを確立し、LISP パケットのカプセル化解除時にフィルタリストをダウンロードして使用します。**members** または **locator-set** キーワードのいずれか、あるいは両方を指定する必要があります。

members キーワードを指定すると、xTR は、登録済みの RLOC メンバーシップリストを自動的に取得するために、設定されたマップサーバとの信頼性の高いトランスポート（TCP）セッションを確立しようとします。**locator-set** が指定されている場合、そのロケータセット内に設定されているロケータに対してフィルタリングが実行されます。locator-set と「members」キーワードの両方が指定されている場合は、設定されているロケータと自動的に検出されたロケータがマージされ、その結果のリストがカプセル化が解除されたパケットに使用されます。

（注）
- (P)xTR は通常、複数のマップサーバと通信します。ただし、すべての信頼性の高いトランスポートセッションがダウンした場合、既存の（失効している可能性がある）フィルタリストが短期間（数分）使用されたままになります。その間、(P)xTR は MS を使用してセッションを再確立してメンバーシップを更新しようと試みます。
- フィルタリストをダウンロードできない場合、または既存のリストがタイムアウトになった場合、パケットはドロップされます（フェールクローズ）。
- xTR が（DHCP を介してなどで）RLOC を変更した場合は、RLOC が変更されるとすぐに、マップサーバへの登録が更新され、新しい登録済み RLOC がこの IID/VPN のすべての「メンバ」にプッシュされます（イベント駆動）。

始める前に

次の前提条件を満たしていることを確認してください。

- xTR では、TCP ベースの信頼性の高いトランスポートセッションは、UDP ベース（通常）の マップ登録プロセスが正常に完了した後にのみ確立されます。
- PxTR では、このデバイスは（通常は）マップサーバに登録されないため、信頼性の高いトランスポートセッションの確立とフィルタリストのダウンロードを可能にするために、「スタブ」（偽）のマップ登録設定を追加する必要があります。マップサーバでは、このセッションの確立を許可するために、map-server rloc members modify-discovered add コマンドに PETR RLOC を含める必要があります。

手順の概要

1. **configure**
2. **router lisp**
3. **exit**
4. **locator-set** *name IP_address*
5. **eid-table** { **default** | [**vrf** *vrf_name*]} **instance-id** *instance_id*
6. **address-family** { **ipv4** | **ipv6** } **unicast**
7. **etr map-server** *IP_address* { **key** [**clear** | **encrypted**] **LINE** | **proxy-reply** }
8. **itr map-resolver** *map-resolver-address*
9. **map-cache** *destination-EID-prefix / prefix-length* { **action** { **drop** | **map-request** | **native-forward**} | **locator** *locator-address* **priority** *priority_value* | **weight** *weight_value*
10. **database-mapping** *EID-prefix/prefixlength locator* **locator-set site priority** *priority* **weight** *weight*
11. **exit**
12. **decapsulation filter rloc source** [**locator-set** *locator_set_name*][**members**]
13. **locator-table** *name* [**default** | **vrf** *vrf_name*]
14. **commit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure**	
ステップ 2	**router lisp** 例： `RP/0/RSP0/cpu 0: router(config)# router lisp`	指定したルーティングインスタンスの LISP を有効にし、ルータを Locator and ID Separation Protocol（LISP）コンフィギュレーションモードにします。
ステップ 3	**exit** 例： `RP/0/RSP0/cpu 0: routerRP/0/0/CPU0:ios(config-lisp-afi)#exit`	ルータを LISP コンフィギュレーションモードに戻します。
ステップ 4	**locator-set** *name IP_address* 例： `RP/0/RSP0/cpu 0: router(config-lisp)#locator-set loc_sh1_vrf1 202.1.0.1`	名前付きのロケータセットサイトを設定し、ループバックまたはその他の出力トンネルルータ（ETR）のインターフェイスの RLOC IP アドレスを指定します。
ステップ 5	**eid-table** { **default** | [**vrf** *vrf_name*]} **instance-id** *instance_id* 例： `RP/0/RSP0/cpu 0: router(config-lisp)#eid-table default instance-id <IID-A>`	デフォルト（グローバル）のルーティングテーブルか、または指定した VRF を選択して、設定したインスタンス ID と関連付けます。
ステップ 6	**address-family** { **ipv4** | **ipv6** } **unicast** 例： `RP/0/RSP0/cpu 0: router(config-lisp-afi)# address-family ipv4 unicast`	IPv4 または IPv6 アドレスファミリを指定して、アドレス ファミリ コンフィギュレーション モードを開始します。 • この例では、ユニキャスト IPv4 アドレスファミリを指定します。
ステップ 7	**etr map-server** *IP_address* { **key** [**clear** | **encrypted**] **LINE** | **proxy-reply** } 例： `RP/0/RSP0/cpu 0: router(config-lisp-afi)#etr map-server 204.1.0.1 key encrypted lisp`	ロケータや認証キーなどの etr map-server（MS）に関連するオプションを指定します。
ステップ 8	**itr map-resolver** *map-resolver-address* 例： `RP/0/RSP0/cpu 0: router(config-lisp-afi)#itr map-resolver 204.1.0.1`	IPv4 EID から RLOC へのマッピングを解決するための ITR マップ要求で使用されるように LISP マップリゾルバの IPv4 または IPv6 のロケータアドレスを設定します。

	コマンドまたはアクション	目的

ステップ 9	**map-cache** *destination-EID-prefix / prefix-length* { **action** { **drop** | **map-request** | **native-forward**} | **locator** *locator-address* **priority** *priority_value* | **weight** *weight_value*
	例：
	`RP/0/RSP0/cpu 0: router(config-lisp-afi)#map-cache 12.2.0.0/24 map-request`
	`RP/0/RSP0/cpu 0: router(config-lisp-afi)#map-cache 102.2.0.0/24 map-request`
	`RP/0/RSP0/cpu 0: router(config-lisp-afi)#map-cache 103.2.0.0/24 map-request`
		静的 IPv4 EID から RLOC、または静的 IPv6 EID から RLOC のマッピング関係とそれに関連付けられたトラフィックポリシーを設定するか、あるいは宛先 IPv4 EID プレフィックスまたは宛先 IPv6 EID プレフィックスと関連付けられたパケット処理動作を静的に設定します。

ステップ 10	**database-mapping** *EID-prefix/prefixlength locator* **locator-set site priority** *priority* **weight** *weight* 例： `RP/0/RSP0/cpu 0: router(config-lisp-afi)#database-mapping 11.2.0.0/24 201.1.0.1 priority 1 weight 100`	この LISP サイトの EID-to-RLOC のマッピング関係と、それに関連するトラフィック ポリシーを設定します。 （注） この eid-table vrf での EID-to-RLOC のすべてのマッピングおよび LISP サイトのインスタンス ID が設定されるまで、この手順を繰り返します。
ステップ 11	**exit** 例： `RP/0/RSP0/cpu 0: router(config-lisp-afi)#exit`	ルータを LISP コンフィギュレーション モードに戻します。
ステップ 12	**decapsulation filter rloc source** [**locator-set** *locator_set_name*][**members**] 例： `RP/0/RSP0/cpu 0: router(config-lisp)#decapsulation filter rloc source member locator-set loc_sh1_vrf1`	送信元 RLOC ベースのカプセル化解除フィルタリング機能を有効にします。 • **members** キーワードを使用すると、設定されたマップサーバとの信頼性の高いトランスポート（TCP）セッションの確立、およびマップサーバが保持しているカプセル化解除フィルタリストのダウンロードが可能になります。 • **locator-set** キーワードが使用され、単独で含まれていた場合は locator-set で指定されたプレフィックスが使用されます。member キーワードと組み合わせて使用されている場合は（ダウンロードされた）ダイナミックリストに追加されます。

	コマンドまたはアクション	目的

ステップ 13	**locator-table** *name* [**default** | **vrf** *vrf_name*]
	例：
	`RP/0/RSP0/cpu 0:`
	`router(config-lisp)#locator-table vrf 1`
		ルーティングロケータのアドレス空間がルータの Locator/ID Separation Protocol（LISP）のインスタンス化に到達可能な Virtual Route Forwarding（VRF）テーブルを関連付けます。

ステップ 14	**commit**	

例

この例では、204.1.0.1 のマップサーバとの信頼性の高いトランスポートセッションを確立し、カプセル化解除フィルタリスト（この場合は IID 1002）をダウンロードし、カプセル化解除の前にこのフィルタリストを使用してすべての LISP カプセル化パケットの送信元チェックを行うように xTR が設定されます。

```
router lisp
 address-family ipv4 unicast
 !
 locator-set loc_sh1_vrf1
   202.1.0.1
   203.1.0.1
 !
 eid-table vrf sh1_vrf2 instance-id 1002
  address-family ipv4 unicast
   etr map-server 204.1.0.1 key encrypted lisp
   etr
   itr map-resolver 204.1.0.1
   itr
   map-cache 12.2.0.0/24 map-request
   map-cache 102.2.0.0/24 map-request
   map-cache 103.2.0.0/24 map-request
   database-mapping 11.2.0.0/24 201.1.0.1 priority 1 weight 100
   database-mapping 101.2.0.0/24 201.1.0.1 priority 1 weight 100
  !
  decapsulation filter rloc source member locator-set
     loc_sh1_vrf1
 !
 locator-table default
```

カプセル解除化フィルタリストの作成、保持、および配布

マップサーバは、サイト コンフィギュレーション モードで map-server rloc members distribute コマンドを使用して、適切な LISP デバイスに対し、インスタンス ID 単位でカプセル化解除フィルタリストを動的に作成、保持、および配布するように設定できます。設定されている場合は次のようになります。

- マップサーバは、適切な xTR との TCP ベース LISP の信頼性の高いトランスポートセッションを確立できるようにします。

- マップサーバは、登録された LISP サイトの RLOC アドレスに基づいて、LISP サイトの RLOC のリストを作成または保持（IID 単位）します。
- マップサーバは、信頼性の高いトランスポートメカニズムを介してフィルタを確立済みのデバイスにプッシュするか、または更新します。

（注）

- データプレーンセキュリティは、「map-server roc members distribute」コマンドを使用して有効になります。動的に保持されている RLOC フィルタリストに追加するか、または上書きするには、オプションコマンドの「map-server rloc members modified-discovered [add | override]」を使用します。
- この機能はカプセル化解除を実行している (P)xTR デバイスに設定されている decapsulation filter rloc source コマンドと組み合わせて使用されます。

次に、特定の LISP サイトとの信頼性の高いトランスポートセッションを作成し、カプセル化解除フィルタリストを動的に作成、保持、配布するようにマップサーバを設定する例を示します。

```
router lisp
 locator-set PxTR_set
  2001:DB8:E:F::2
  exit
 !
 eid-table vrf 1001 instance-id 1001
  map-server rloc members modify-discovered add locator-set PxTR_set
  exit
 !
---<skip>---
 !
 map-server rloc members distribute
!
```

カプセル化解除フィルタリストの追加または上書き

カプセル化解除フィルタリストを動的に作成、維持、配布するようにマップサーバが設定されている場合は、EID テーブル設定モードで map-server rloc members modify-discovered コマンドを使用することでカプセル化解除フィルタリストを追加または上書きできます。以下を使用できます。

- PxTR がアーキテクチャに含まれている場合、PITR LISP はパケットを ETR にカプセル化します。そのため、ETR には、そのカプセル化解除フィルタリスト内に PITR RLOC を含める必要があります。PITR はマップサーバに登録されないため、それらの RLOC はカプセル化解除フィルタリストに自動的には含められません。そのため、このコマンドを使用し、設定を介して追加する必要があります。
- また、PETR は、カプセル化解除時にフィルタ処理するように設定することもできますが、PETR はマップサーバに登録されないため、カプセル化解除フィルタリストを取得する手段が必要です。このコマンドの **add** 形式には、PETR でカプセル化解除フィルタリストを

取得するための信頼性の高いトランスポートセッションをマップサーバと確立するメカニズムが含まれています。

- 診断/トラブルシューティング上の理由から、カプセル化解除フィルタリスト全体を（一時的に）上書きすると便利な場合があります。

（注） カプセル化解除フィルタリストを取得するために、PETR がマップサーバに「偽の登録」を行えるよう、add 関数を含める必要があります。このコマンドに PETR RLOC を挿入すると、PETR は信頼性の高いトランスポートセッションを確立できます。

この例では、特定の LISP サイトと信頼性の高いトランスポートセッションを作成するようにマップサーバを設定し、カプセル化解除フィルタリストを作成、維持、配布します。また、静的に設定された PxTR IPv6 RLOC のアドレス（2001:db8:e:f::2）を使用して、動的に作成されたフィルタリスト（登録したサイトの RLOC アドレスで構成）を変更するようにも設定されています。

```
router lisp
 locator-set PxTR_set
  2001:DB8:E:F::2
  exit
 !
 eid-table vrf 1001 instance-id 1001
  map-server rloc members modify-discovered add locator-set PxTR_set
  ipv4 route-export site-registration
  exit
 !
---<skip>---
 !
 map-server rloc members distribute
!
```

LISP TCP の信頼性の高いトランスポートセッションのリセット

xTR と MS の間で LISP TCP の信頼性のトランスポートセッションをリセットするには、**clear lisp vrf** コマンドを EXEC モードで使用します。

手順の概要

1. **clear lisp vrf** *VRF_name* **session** {*peer_address* | *****

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**clear lisp vrf** *VRF_name* **session** {*peer_address* | ***** 例： `RP/0/0/CPU0:ios#clear lisp vrf test session *`	ピアアドレスを指定すると、そのピアへの TCP 接続がクリアされます。「*」オプションを指定すると、すべての LISP の信頼性の高いトランスポートセッションがクリアされます。

データプレーンのセキュリティ設定の確認

データプレーンのセキュリティ設定を確認するには、次のタスクを実行します。

手順の概要

1. **show lisp session**
2. **show lisp site** [**instance-id** *EID instance-ID*] **rloc members** [**registrations** [*rloc-addr*]]
3. **show lisp vrf** *vrf_name* **session** [*peer_address*]
4. **show lisp decapsulation filter**
5. **show cef vrf** [*locator-vrf*] *address_family* **lisp decapsulation** [**instance-id** *EID-instance-ID*] **detail location** *RLOC-facing LC*
6. **show controllers np struct LISP-INSTANCE-HASH detail all-entries** ［ **all** ｜*np* ］**location** *RLOC-facing LC*

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**show lisp session**
	例：
	`R11-MSMR#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`
	`R11-MSMR#`
		LISP デバイスで、信頼性の高いトランスポート（TCP） セッションの現在のリストを表示するには、EXEC コンフィギュレーション モードで **show LISP session** コマンドを使用します。この例では、信頼性の高いトランスポート LISP セッションがマップサーバに表示されます。出力では、7つのセッションが確立され、1 つのセッションが Init 状態になっています（decapsulation filter rloc source member コマンドがそのサイトに適用されておらず、セッションは確立されませんでした）。

ステップ 2	**show lisp site** [**instance-id** *EID instance-ID*] **rloc members** [**registrations** [*rloc-addr*]]
	例：
	`R114-MSMR#show lisp site rloc members`
	`LISP RLOC membership for EID table default (IID 0), 5 entries`
	
	`RLOC Origin Valid`
		収集および設定された EID インスタンスのメンバーシップを表示するには、EXEC コンフィギュレーション モードで **show lisp site** コマンドを使用します。出力の 「origin」列には、RLOC メンバが手動で設定されたのか、または受信した登録から自動的に収集されたのか、あるいはその両方かが示されます。「valid」列には、RLOC が (P)xTR に配布される有効なメンバであるかどうかが示されます。リストされ

	コマンドまたはアクション	目的
	1.2.3.4 config Yes 10.0.1.2 registration Yes 10.0.2.2 config & registration Yes 13:12::1 config Yes 2001:DB8:2:3::2 registration Yes	ているRLOCは、それが登録から収集されたものであっても、「modify-discovered」設定に「override」オプションが使用されており、指定されたロケータセットにそのRLOCが含まれていない場合は有効でない可能性があります。オプションの「registrations」キーワードが指定されている場合、このコマンドはメンバーシップエントリに関与する登録のリストを表示します。
ステップ3	**show lisp vrf** *vrf_name* **session** [*peer_address*] 例： On xTR: RP/0/RSP1/CPU0:VKG-1#sh lisp vrf default session Sessions for VRF default, total: 1, established: 1 Peer State Up/Down In/Out Users 204.1.0.1 Up 06:49:05 0/1 1 RP/0/RSP1/CPU0:VKG-1# On MSMR: sh lisp vrf default session Sessions for VRF default, total: 2, established: 2 Peer State Up/Down In/Out Users 201.1.0.1 Up 06:48:49 1/0 0 202.1.0.1 Up 06:48:36 2/0 0 RP/0/RSP0/CPU0:VKG-4#	
ステップ4	**show lisp decapsulation filter** 例： RP/0/RSP0/CPU0:lisp9-a9k-1#show lisp eid-table se2 decapsulation filter LISP decapsulation filter for EID table vrf se2 (IID 16777212), 5 entries Source RLOC Added by 22:22::10 MS 190::190 33:33::20 MS 190::190 88:88::30 MS 190::190 99:99::30 MS 190::190 110:110::40 MS 190::190 RP/0/RSP0/CPU0:lisp9-a9k-1#	LISP デバイスで、選択した送信元 RLOC のカプセル化解除フィルタに関連するデータを表示するには、EXEC コンフィギュレーション モードで **show lisp decapsulation filter** コマンドを使用します。この例では、カプセル化解除フィルタ情報は (P)xTR for Instance-ID (IID 16777212 に表示されます。この出力では、5 つの送信元 RLOC アドレスが定義されており、このリストのすべてのメンバがマップサーバとの信頼性の高いトランスポートセッションによって提供されるリスト内で定義されています。

	コマンドまたはアクション	目的

ステップ 5	**show cef vrf** [*locator-vrf*] *address_family* **lisp decapsulation** [**instance-id** *EID-instance-ID*] **detail location** *RLOC-facing LC*
	例：
	`RP/0/RSP0/CPU0:lisp9-a9k-1#show cef ipv6 lisp decapsulation instance-id 16777212 loc 0/0/cpu0`
	
	`Number of EID tables handling LISP payload received in this table: 3`
	
	`Transport LISP ipv6 packets received in VRF: default, Instance ID: 16777212`
	`Payload IPv4 is : decapsulated`
	`Payload IPv6 is : decapsulated`
	`Payload switched in VRF : se2 (0xe00000 1d /0xe080001d)`
	`H/W driver signalled : active`
	`Binding in retry : no`
	
	`Source RLOC Prefix Filter : enabled`
	`Lookup statistics (s/w) :`
	`Misses : 8`
	`Matches (historic) : 0`
	`H/W driver signalled : active`
	`Binding in retry : no`
	`Platform space : allocated`
	
	`Len Prefix Action Matches Attributes`
	
	`128 22:22::10 accept 0 h/w [active, plt space]`
	`128 33:33::20 accept 0 h/w [active, plt space]`
	`128 88:88::30 accept 0 h/w [active, plt space]`
	`128 99:99::30 accept 0 h/w [active, plt space]`
	`128 110:110::40 accept 0 h/w [active, plt space]`
		この例では、カプセル化解除フィルタのサマリー情報が (P)xTR に表示されます。

ステップ 6	**show controllers np struct LISP-INSTANCE-HASH detail all-entries** [**all** | *np*] **location** *RLOC-facing LC*
	例：
	`RP/0/RSP0/CPU0:lisp9-a9k-1#show controllers np struct LISP-INSTANCE-HASH detail all-entries np0 location 0/0/cpu0`
	
	`Node: 0/0/CPU0:`
	`----------`
	
	`NP: 0`
	`Struct 114: LISP_INSTANCE_HASH_STR (maps to uCode Str=79)`
	`Struct is a LOGICAL entity inside a shared`	

	コマンドまたはアクション	目的
	PHYSICAL resource Reserved Entries: Logical 0, Physical 0 Used Entries: Logical 79, Physical 79 Max Entries: Logical 86016, Physical 86016 Entries Shown: Logical 79 ---------- Entry 1: >> Key: 4affffe 00000099 00990000 00000000 00000000 0030 Size: 22 Mask: ffffffff ffffffff ffffffff ffffffff ffffffff ffff Size: 22 Result: 51000000 1e000000 Size: 8 Entry 2: >> Key: 4976adf1 1c005858 0e1e0000 00000000 00000000 0000 Size: 22 Mask: ffffffff ffffffff ffffffff ffffffff ffffffff ffff Size: 22 Result: 51000000 1c000000 Size: 8 Entry 3: >> Key: 4976adf1 1c006e6e 0e280000 00000000 00000000 0000 Size: 22 Mask: ffffffff ffffffff ffffffff ffffffff ffffffff ffff Size: 22 Result: 51000000 1c000000 Size: 8 Entry 4: >> Key: 41000000 20002121 0b140000 00000000 00000000 0000 Size: 22 Mask: ffffffff ffffffff ffffffff ffffffff ffffffff ffff Size: 22 Result: 51000000 1f000000 Size: 8 Entry 5: >> Key: 4**affffc** 0000**0099 00990000 00000000 00000000 0030** Size: 22 Mask: ffffffff ffffffff ffffffff ffffffff ffffffff ffff Size: 22 Result: 51000000 **1d**000000 Size: 8 Entry 6: >> Key: 4a0003e8 19000033 00330003 00000000 00000000 0020 Size: 22 Mask: ffffffff ffffffff ffffffff ffffffff ffffffff ffff Size: 22 Result: 51000000 19000000 Size: 8 Entry 7: >> <Snipped> End NP Show Structure Display	

その他の参考資料

以降の項では、LISP の実装に関する関連資料について説明します。

関連資料

関連項目	マニュアル タイトル
LISP コマンド：コマンドシンタックスの詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上の注意事項、および例	*Routing Command Reference for Cisco ASR 9000 Series Routers*
LISP コンフィギュレーション ガイド、Cisco IOS リリース	『IP Routing: LISP Configuration Guide, Cisco IOS Release 15M&T』

標準

標準	タイトル
draft-kouvelas-lisp-reliable-transport-00.txt	『*LISP Reliable Transport*』（C. Cassar、I. Kouvelas、および D. Lewis）
RFC 6830	*Locator/ID Separation Protocol*（*LISP*）
RFC 6832	*Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites*
RFC 6833	*Locator/ID Separation Protocol (LISP) Map-Server Interface*

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/techsupport