Cisco Nexus 9000 シリーズ NX-OS IP SLA 構成ガイド、リリース 10.3(x)

初版：2022 年 8 月 19 日

最終更新：2023 年 2 月 15 日

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

目 次

はじめに

この前書きは、次の項で構成されています。

対象読者

このマニュアルは、Cisco Nexus スイッチの設置、設定、および維持に携わるネットワーク管理者を対象としています。

表記法

コマンドの説明には、次のような表記法が使用されます。

表記法	説明
bold	太字の文字は、表示どおりにユーザが入力するコマンドおよびキーワードです。
italic	イタリック体の文字は、ユーザが値を指定する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角かっこで囲んで示しています。
[x | y]	いずれか1つを選択できる省略可能なキーワードや引数は、角かっこで囲み、縦棒で区切って示しています。
{x | y}	必ずいずれか1つを選択しなければならない必須キーワードや引数は、波かっこで囲み、縦棒で区切って示しています。

表記法	説明
[x {y | z}]	角かっこまたは波かっこが入れ子になっている箇所は、任意または必須の要素内の任意または必須の選択肢であることを表します。角かっこ内の波かっこと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。
`variable`	ユーザが値を入力する変数であることを表します。イタリック体が使用できない場合に使用されます。
string	引用符を付けない一組の文字。string の前後には引用符を使用しないでください。引用符を使用すると、その引用符も含めて string と見なされます。

例では、次の表記法を使用しています。

表記法	説明
`screen フォント`	スイッチが表示する端末セッションおよび情報は、スクリーンフォントで示しています。
太字の screen フォント	ユーザが入力しなければならない情報は、太字の screen フォントで示しています。
イタリック体の *screen* フォント	ユーザが値を指定する引数は、イタリック体の screen フォントで示しています。
<>	パスワードのように出力されない文字は、山カッコ（<>）で囲んで示しています。
[ ]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

Cisco Nexus 9000 シリーズ スイッチの関連資料

Cisco Nexus 9000 シリーズ スイッチ全体のマニュアル セットは、次の URL にあります。

http://www.cisco.com/en/US/products/ps13386/tsd_products_support_series_home.html

マニュアルに関するフィードバック

このマニュアルに関する技術的なフィードバック、または誤りや記載もれなどお気づきの点がございましたら、HTML ドキュメント内のフィードバック フォームよりご連絡ください。ご協力をよろしくお願いいたします。

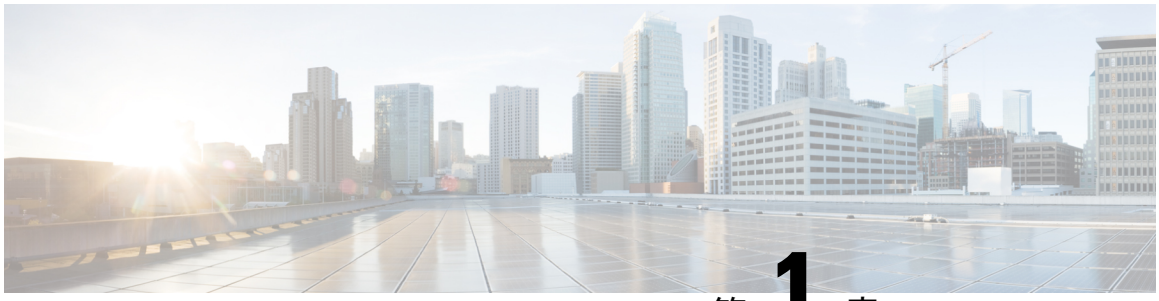
通信、サービス、およびその他の情報

- シスコからタイムリーな関連情報を受け取るには、Cisco Profile Manager でサインアップしてください。
- 重要な技術によりビジネスに必要な影響を与えるには、シスコサービスにアクセスしてください。
- サービス リクエストを送信するには、シスコサポートにアクセスしてください。
- 安全で検証済みのエンタープライズクラスのアプリケーション、製品、ソリューション、およびサービスを探して参照するには、Cisco Marketplace にアクセスしてください。
- 一般的なネットワーク、トレーニング、認定関連の出版物を入手するには、Cisco Press にアクセスしてください。
- 特定の製品または製品ファミリの保証情報を探すには、Cisco Warranty Finder にアクセスしてください。

Cisco バグ検索ツール

Cisco バグ検索ツール（BST）は、シスコ製品とソフトウェアの障害と脆弱性の包括的なリストを管理する Cisco バグ追跡システムへのゲートウェイとして機能する、Web ベースのツールです。BST は、製品とソフトウェアに関する詳細な障害情報を提供します。

通信、サービス、およびその他の情報



第 1 章

新機能と変更情報

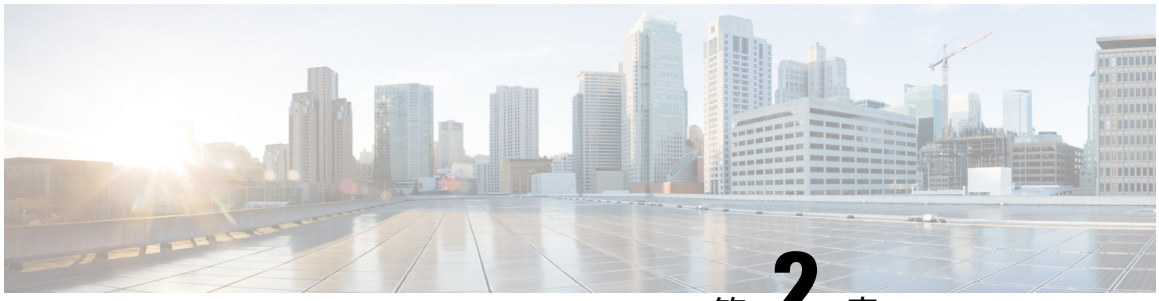
新機能と変更情報

次の表は、*Cisco Nexus 9000* シリーズ *NX-OS IP SLA* 構成ガイド リリース *10.1(x)* に記載されている新機能および変更機能をまとめたものです。それぞれの説明が記載されている箇所も併記されています。

表 *1*: 新機能および変更された機能

機能	説明	変更が行われたリリース	参照先
NA	このリリースで追加された新機能はありません。	10.3(1)F	該当なし

新機能と変更情報



第 2 章

IP SLA の概要

この章では、Cisco NX-OS IP サービス レベル契約（SLA）の概要について説明します。

Cisco NX-OS IP SLA に関する情報

多くの企業ではビジネスのほとんどをオンラインで行い、サービスの損失は企業の収益性に影響を及ぼすことがあります。今では、インターネット サービス プロバイダー（ISP）や内部 IT 部門でさえも、定義済みのサービス レベル（サービス レベル契約）を提供して、お客様に一定の予測可能性を提供しています。

ビジネス クリティカルなアプリケーション、Voice over IP（VoIP）ネットワーク、音声および表示による会議、マルチプロトコル ラベル スイッチング（MPLS）、およびバーチャル プライベート ネットワーク（VPN）の最新のパフォーマンス要件により、企業内では、パフォーマンス レベルに合わせた統合 IP ネットワークの最適化が求められています。ネットワーク管理者にとっては、アプリケーション ソリューションを支えるサービス レベル契約をサポートする必要性がますます高まっています。IP サービス レベル契約（SLA）を使用すると、IP アプリケーションおよび IP サービスの IP サービス レベルを管理できます。

Cisco NX-OS IP SLA は、アクティブ トラフィック モニタリングを使用します。これにより、継続的で信頼性のある予測可能な方法でトラフィックが生成され、ネットワーク パフォーマンスを測定できます。Cisco NX-OS IP SLA はネットワークにデータを送信し、複数のネットワーク ロケーション間あるいは複数のネットワーク パス内のパフォーマンスを測定します。ネッ

トワーク データおよびIP サービスをシミュレーションし、ネットワーク パフォーマンス情報をリアル タイムで収集します。収集される情報には、応答時間、一方向遅延、ジッター（パケット間の遅延のばらつき）、パケット損失、音声品質スコアリング、ネットワーク リソースの可用性、アプリケーションのパフォーマンス、およびサーバーの応答時間に関するデータが含まれます。Cisco NX-OS IP SLA はトラフィックを生成、分析して、Cisco NX-OS デバイス間または Cisco NX-OS デバイスからネットワーク アプリケーション サーバーのようなリモート IP デバイスへのパフォーマンスを測定することにより、アクティブ モニタリングを実行します。Cisco NX-OS IP SLA のさまざまな動作による測定統計情報を、トラブルシューティング、問題分析、ネットワーク トポロジの設計に使用できます。

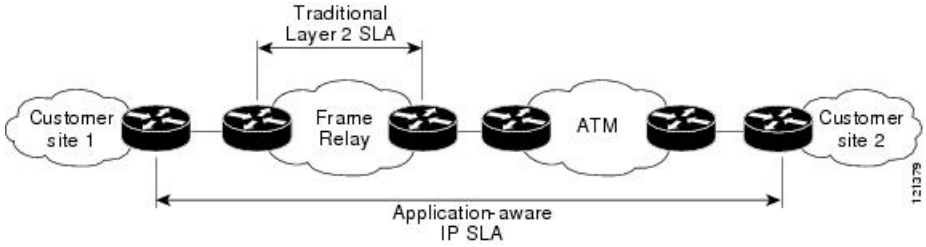
（注） IPSLA はロールバックをサポートしていません。ロールバックは、CLI を介した IPSLA 構成に関連しています。

Cisco NX-OS P SLA では、従来のサービス レベル契約と比べて次のような改善を実現できます。

- エンドツーエンド測定：ネットワークの端からもう一方の端までパフォーマンスを測定できることにより、エンドユーザによるネットワーク利用状況をより広い到達範囲でより正確に表現できます。
- 詳細化：遅延、ジッター、パケット シーケンス、レイヤ 3 接続、パスとダウンロード時間などの双方向のラウンドトリップの数値に詳細化される統計情報により、レイヤ 2 リンクの帯域幅だけよりも詳細なデータが得られます。
- 展開の簡易化：Cisco IOS IP SLA は、大きいネットワーク内で既存のシスコ デバイスを活用することにより、従来のサービス レベル契約で必要になることの多い物理的なプローブよりも、簡単かつ低コストで実装されます。
- アプリケーション認識型モニタリング：Cisco NX-OS IP SLA は、レイヤ 3 からレイヤ 7 で実行されているアプリケーションによって生成されたパフォーマンス統計情報をシミュレートし、測定できます。従来のサービス レベル契約では、レイヤ 2 パフォーマンスしか測定できません。
- 広範囲：Cisco NX-OS IP SLA のサポートは、ローエンド スイッチからハイエンド スイッチまでのシスコ ネットワーキング デバイスに含まれています。この幅広い展開により、Cisco NX-OS IP SLA は、従来のサービス レベル契約よりも高い柔軟性を備えています。

次の図に、アプリケーションのサポートも含め、エンドツーエンドのパフォーマンス測定をサポートするために、Cisco NX-OS IP SLA がどのように従来のレイヤ 2 サービス レベル契約の概念を取り込み、より広い範囲に適用されているかを示します。

図 *1* : 従来のサービス レベル契約と *Cisco IOS IP SLA* の範囲の比較



Cisco NX-OS IP SLA を使用して、サービス レベル契約を測定、提供、確認できます。また、IP サービスおよび IP アプリケーションのネットワーク パフォーマンスを分析してトラブルシューティングを行えます。Cisco NX-OS IP SLA の特定の動作に応じて、遅延、パケット損失、ジッター、パケット シーケンス、接続、パス、サーバーの応答時間、およびダウンロード時間の統計情報がシスコ デバイス内でモニタでき、CLI および SNMP MIB の両方に保存できます。パケットには設定可能な IP レイヤ オプションとアプリケーション層オプションがあります。たとえば、送信元および宛先の IP アドレス、ユーザー データグラム プロトコル（UDP）/TCP ポート番号、サービス タイプ（ToS）バイト（Diffserv コードポイント（DSCP）および IP プレフィックス ビットを含む）、バーチャル プライベート ネットワーク（VPN）ルーティング/転送インスタンス（VRF）、URL Web アドレスなどが設定できます。

Cisco NX-OS IP SLA には、SNMP を使用してアクセスできるため、CiscoWorks Internet Performance Monitor（IPM）のようなパフォーマンス モニタリング アプリケーションや他のサードパーティ製のシスコ パートナー パフォーマンス管理製品からも使用できます。

Cisco NX-OS IP SLA 動作によって収集されたデータに基づく SNMP 通知により、パフォーマンスが指定したレベルを下回った場合や問題が修正された場合に、ルータはアラートを受信できます。Cisco NX-OS IP SLA は、外部ネットワーク管理システム（NMS）アプリケーションとシスコ デバイス上で実行されている Cisco NX-OS IP SLA 動作との間のインタラクションに Cisco RTTMON MIB を使用します。Cisco NX-OS IP SLA 機能から参照されるオブジェクト変数の詳細については、Cisco MIB Web サイトから入手できる CISCO-RTTMON-MIB.my ファイルのテキストを参照してください。

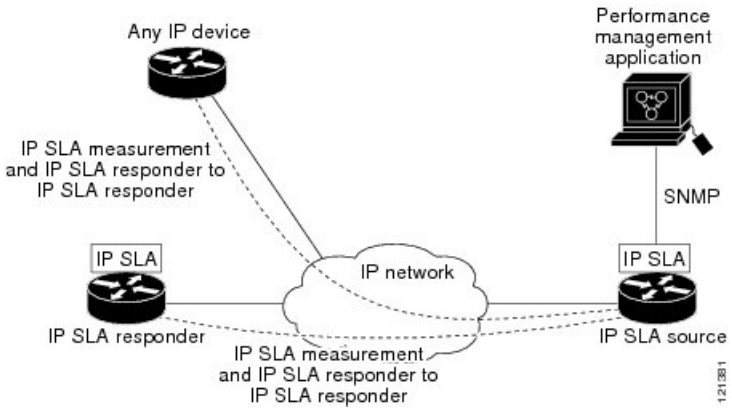
Cisco NX-OS IP SLA を使用したネットワーク パフォーマンスの測定

Cisco NX-OS IP SLA を使用して、コア、分散、エッジといったネットワークの任意の領域間のパフォーマンスをモニタできます。モニタリングは、物理的なプローブを展開しなくても、時間と場所を問わず実行できます。

Cisco NX-OS IP SLA は、生成されたトラフィックを使用して、スイッチなどの 2 つのネットワーク デバイス間のネットワーク パフォーマンスを測定します。次の図に、Cisco NX-OS IP SLA デバイスが生成パケットを接続先デバイスに送信したとき、Cisco NX-OS IP SLA がどのように開始されるかを示します。Cisco NX-OS IP SLA 動作のタイプにもよりますが、接続先デバイスはそのパケットを受信した後、送信元でパフォーマンス メトリックを計算できるようにタ

イムスタンプ情報を返信します。Cisco NX-OS IP SLA 動作は、特定のプロトコル（UDP など）を使用してネットワークの送信元から接続先へのネットワーク測定を行います。

図 *2 : Cisco NX-OS IP SLA* 動作



Cisco NX-OS IP SLA ネットワーク パフォーマンス測定を実施するには、次のタスクを実行する必要があります。

1. Cisco IOS IP SLA Responder が有効でない場合は、有効にします。
2. 必要な Cisco NX-OS IP SLA 動作タイプを構成します。
3. 指定された Cisco NX-OS IP SLA 動作タイプに使用可能なオプションを設定します。
4. 必要であれば、しきい値条件を設定します。
5. 動作の実行スケジュールを指定し、しばらく動作を実行して統計情報を収集します。
6. Cisco NX-OS CLI を使用するか、ネットワーク管理システムと SNMP を併用して、動作の結果を表示し、確認します。

Cisco NX-OS IP SLA 動作タイプ

Cisco NX-OS IP SLA 動作には、次のようにさまざまなタイプがあります。

- UDP ジッター
- VoIP 用の UDP ジッタ
- UDP エコー
- 伝送制御プロトコル（TCP）接続
- 複数動作スケジューラ
- 予防的しきい値モニタリング

Cisco NX-OS IP SLA Responder および IP SLA 制御プロトコル

レスポンダは接続先の Cisco ルーティング デバイスに組み込まれたコンポーネントで、システムが Cisco NX-OS IP SLA 要求パケットを予想して応答できるようにします。IP SLA Responder により、専用プローブがなくても正確な測定が可能になります。標準的な ICMP ベースの測定では得られない追加の統計情報も得られます。Cisco NX-OS IP SLA 制御プロトコルは、IP SLA Responder がどのポートで待ち受けと応答を行うかを通知するために使用するメカニズムを提供します。接続先にレスポンダがある場合、送信元にできるのは、Cisco NX-OS デバイスのみです。

IP SLA Responder は、Cisco NX-OS IP SLA 動作から送信されたコントロール プロトコル メッセージを指定されたポートでリッスンします。コントロール メッセージを受信すると、レスポンダは、指定された UDP ポートまたは TCP ポートを、指定された期間、有効状態にします。この間に、レスポンダは要求を受け付け、応答します。レスポンダは、Cisco IOS IP SLA パケットへの応答後、あるいは指定された期間の経過後に、ポートを無効にします。

すべての IP SLA 動作について、IP SLA Responder を宛先デバイスでイネーブルにしなければならないわけではありません。たとえば、接続先スイッチですでに提供されているサービス（Telnet や HTTP など）を選択する場合には、IP SLA Responder を有効にする必要はありません。Cisco 以外のデバイスには、IP SLA Responder を構成できません。この場合、IP SLA はこれらのデバイスにネイティブなサービスに対してのみ、動作パケットを送信できます。

Cisco NX-OS IP SLA 動作のスケジューリング

Cisco NX-OS IP SLA 動作の設定が完了したら、その動作をスケジューリングして、統計情報の取得とエラー情報の収集を開始する必要があります。動作をスケジュールする場合は、すぐに動作を開始するよう指定するか、特定の月、日、時刻に開始するように指定できます。後で動作を開始するように設定する pending オプションもあります。pending オプションは、動作の内部状態の 1 つでもあり、SNMP によって確認できます。トリガーを待機する反応（しきい値）動作の場合も pending オプションを使用します。単一の Cisco IOS IP SLA 動作をスケジューリングすることも、動作のグループを一度にスケジューリングすることもできます。

複数動作のスケジューリングでは、Cisco NX-OS CLI または CISCO RTTMON-MIB により、1 つのコマンドを使用して複数の Cisco NX-OS IP SLA 動作をスケジューリングできます。この機能では、これらの動作を均等な時間間隔で実行するようにスケジューリングすることで、IP SLA モニタリング トラフィックの量を制御できます。このように IP SLA 動作を分散することで、CPU の使用を最小限に抑え、ネットワークの拡張性を向上させることができます。

IP SLA 複数動作のスケジューリング機能の詳細については、「IP SLA 複数動作スケジューラの設定」の項を参照してください。

Cisco NX-OS IP SLA 動作のしきい値モニタリング

サービス レベル契約モニタリングを適切にサポートするには、あるいはネットワーク パフォーマンスを予防的に測定するには、しきい値機能が最も重要になります。信頼性のある一貫した測定を行えば、問題はただちに特定され、トラブルシューティングにかかる時間を短縮できます。サービス レベル契約を展開するには、違反が発生した場合にただちに通知するメカニズムが必要です。Cisco NX-OS IP SLA は次のような場合にイベントによってトリガーされる SNMP トラップを送信できます。

- 接続の損失
- タイムアウト
- RTT しきい値
- 平均ジッターしきい値
- 一方向パケット損失
- 一方向ジッター
- 一方向平均オピニオン評点（MOS）
- 一方向遅延

また、Cisco NX-OS IP SLA しきい値違反により、さらに詳しく分析するために別の Cisco NX-OS IP SLA 動作をトリガーすることができます。

Cisco NX-OS IP SLA 動作のしきい値の使用方法の詳細については、IP SLA 動作の予防的しきい値モニタリングに関する項を参照してください。

MPLS VPN 認識

Cisco NX-OS IP SLA MPLS VPN 認識機能を使用すると、マルチプロトコル ラベル スイッチング（MPLS）仮想プライベート ネットワーク（VPN）内で IP サービス レベルをモニタできます。MPLS VPN 内で IP SLA を使用することにより、サービス プロバイダーは、お客様のサービス レベル契約に従って IP VPN サービスを計画、プロビジョニング、および管理できます。IP SLA 動作は、VPN ルーティングおよび転送（VRF）の名前を指定して、特定の VPN に対して設定できます。

履歴統計情報

Cisco NX-OS IP SLA には、次に示す 3 つのタイプの履歴統計情報が保持されます。

- 集約統計情報：デフォルトでは、IP SLA によって動作ごとに 2 時間の集計統計情報が保持されます。各動作サイクルからの値は、所定の 1 時間以内のすでに利用可能なデータとともに集約されます。IP SLA の拡張履歴機能を使用すると、集約間隔を 1 時間未満にできます。

- 動作スナップショット履歴：IP SLA は、設定可能なフィルタ（すべて、しきい値超過、障害など）と一致する動作インスタンスごとに、データのスナップショットを保持します。データ セット全体が使用可能であり、集約は行われません。
- 分散統計情報：IP SLA は、設定可能な時間間隔にわたり、頻度分布を維持します。IP SLA によって動作が開始されるたびに、履歴バケット数が指定したサイズに一致するまで、または動作のライフタイムが期限切れになるまで、新しい履歴バケットが作成されます。デフォルトでは、IP SLA 動作の履歴は収集されません。履歴を収集する場合は、動作の 1 つまたは複数の履歴エントリが各バケットに格納されます。履歴バケットのラップは行われません。

IP SLA の注意事項と制約事項

IP SLA には、次の注意事項と制約事項があります。

- キーワードが付いている**show**コマンド**internal**はサポートされていません。
- IP SLA は、Cisco NX-OS ロールバック機能をサポートしていません。
- IPv6 での ICMP エコー操作は、Cisco Nexus 9300 および 9500 シリーズ スイッチでサポートされています。
- Cisco Nexus 3232C および 3264Q スイッチは、ポリシーベース ルーティング（PBR）をサポートしていません。
- 一方向遅延（レイテンシ）測定では、マイクロ秒単位の測定はサポートされていません。ミリ秒などの他の測定単位はサポートされています。
- スイッチの再起動など、多数のインターフェイス ステートの変更が同時に発生する状況では、IP SLA トラックが起動するまでに数分かかることがあります。この状況では、大量の収集ドロップが発生していないか確認してください。**sh policy-map interface control-plane** コマンドを実行し、`match exception glean`（一致例外収集）の下でスイッチの定常状態での継続的なドロップまたは違反を探します。回避策としては、**hardware ip glean throttle maximum** をデフォルトの 1000 から 10,000 に増やすことができます。

IP SLA 実装の制限事項

Cisco NX-OS IP SLA の制限には、次のものがあります。

（注） IPv6 は、Cisco NX-OS リリース 7.0(3)I6(1) から利用できます。

- Cisco IOS XR ソフトウェアでサポートされている IP SLA 構成可能操作の最大数は 500 です。
- 動作のスケジューリングで現在検証されている有効なスケール数は次のとおりです。

- UDP エコー動作の最大数は、デフォルトの頻度では 300 動作です。
- UDP ジッター動作の最大数は、デフォルトの頻度では 200 動作です。
- ICMP IPv4 またはIPv6 エコー動作の最大数は、デフォルトの頻度では 300 動作です。
- TCP 接続動作の最大数は、デフォルトの頻度では 100 動作です。

同じ開始時刻に毎秒 10 より多くの操作をスケジューリングすることは、パフォーマンスに影響する可能性があるため、推奨しません。グループ スケジューリング構成を使用することをお勧めします。

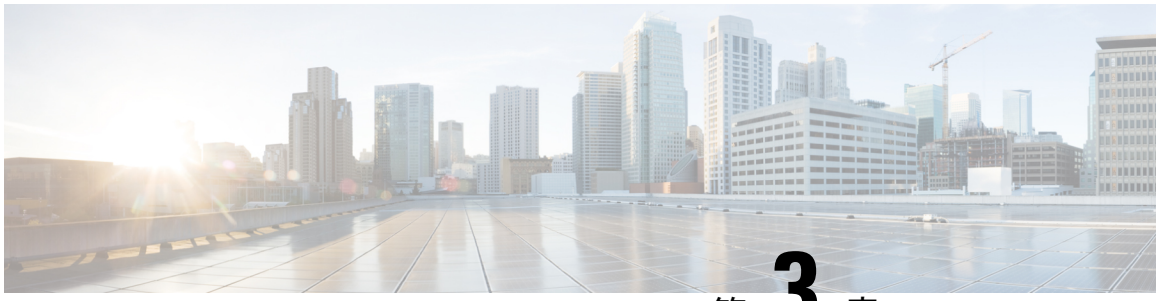
（注） 頻度を 60 秒未満に設定すると、送信されるパケット数が増加します。しかしこのことは、スケジュールされた動作の開始時刻が同じ場合、IP SLA 動作のパフォーマンスに悪影響を与える可能性があります。IP SLA は HA に対応していません。frequency、timeout、および threshold コマンドを構成する前に、次の注意事項を検討してください。

UDP および ICMP ジッター操作の場合は、次のガイドラインに従うことを推奨します。

- frequency > timeout + 2 秒 + num_packets * packet_interval timeout > rtt_threshold num_packet > loss_threshold

他のすべての IP SLA 動作の場合：

- frequency > timeout > rtt_threshold のガイドラインが推奨されます。



第 3 章

IP SLA UDP ジッター動作の設定

この章では、IP サービス レベル契約（SLA）UDP ジッター動作を設定して、IPv4 ネットワークで UDP トラフィックを伝送するネットワークのラウンド トリップ遅延、一方向遅延、一方向ジッター、一方向パケット損失、および接続を分析する方法について説明します。この章では、UDP ジッター動作を使用して収集されたデータを Cisco ソフトウェア コマンドを使用して表示および分析する方法についても説明します。

この章は、次の項で構成されています。

IP SLA UDP ジッタ動作に関する情報

IP SLA UDP ジッター動作では、Voice over IP（VoIP）、Video over IP、またはリアルタイム会議などのリアルタイム トラフィックのアプリケーションのネットワーク適合性を診断することができます。

ジッターとは、パケット間の遅延のばらつきを意味します。複数のパケットが発信元から宛先に連続的に送信された場合、たとえば 10 ms 間隔で送信された場合、ネットワークが理想的に動作していれば、宛先は 10 ms 間隔でパケットを受信します。しかし、ネットワーク内に遅延（キューイング、代替ルートを介した受信など）が存在する場合、パケットの到着間隔は、10 ミリ秒より大きくなる場合も、10 ミリ秒より小さくなる場合もあります。この例を使用すると、正のジッタ値は、パケットの到着間隔が 10 ミリ秒を超えていることを示します。パケットが 12 ms 間隔で到着する場合、正のジッターは 2 ms です。パケットが 8 ms 間隔で到着する場合、負のジッターは 2 ms です。VoIP など遅延に影響されやすいネットワークの場合、正のジッター値は望ましくなく、0 のジッター値が最適です。

しかし、IP SLA UDP ジッター動作の機能は、ジッタのモニタリングだけではありません。UDP ジッター動作には IP SLA UDP 動作によって返されたデータが含まれているため、UDP ジッター動作は多目的データ収集動作に使用できます。IP SLA が生成するパケットは、シーケン

ス情報を送受信するパケット、および送信元および動作ターゲットからのタイムスタンプを送受信するパケットを搬送します。UDP ジッター動作では、以下を測定できます。

- 方向別ジッター（送信元から宛先へ、宛先から送信元へ）
- 方向別パケット損失
- 方向別遅延（一方向遅延）
- ラウンドトリップ遅延（平均 RTT）

データの送信と受信でパスが異なることがあるので（非対称）、方向別データを使用してネットワークの輻輳などの問題が発生している場所を簡単に特定できます。

UDP ジッター動作は、合成（シミュレーション）UDP トラフィックを生成して機能します。UDP ジッター動作は、指定された頻度 F で、送信元スイッチからターゲット スイッチに、サイズ S の N 個の UDP パケットを T ミリ秒間隔で送信します。デフォルトでは、ペイロードサイズが 10 バイト（S）のパケット 10 個（N）を 10 ミリ秒（T）ごとに生成し、60 秒（F）ごとに動作を繰り返します。これらのパラメータはそれぞれ、次の表に示すように、ユーザーが設定できます。

表 *2: UDP* ジッター動作パラメータ

UDP ジッター動作パラメータ	**Default**	**コマンド**
パケット数（n）	10 パケット	**udp-jitter** コマンド、**numpackets** オプション
パケットあたりのペイロードサイズ（S）	32 バイト	**request-data-size** コマンド
パケット間隔（ミリ秒単位）（T）	20 ミリ秒	**udp-jitter** コマンド、interval オプション
動作を繰り返すまでの経過時間（秒単位）（F）	60 秒	**frequency**（**IP SLA**）コマンド

IP SLA UDP ジッター動作を構成するための前提条件

IP SLA UDP ジッター動作を構成するための前提条件は次のとおりです。

- 一方向遅延を正確に測定するには、NTP などによる送信元デバイスとターゲット デバイスとの間のクロック同期が必要です。一方向ジッターおよびパケット損失を測定する場合は、クロック同期は不要です。送信元デバイスとターゲットデバイスの間でクロックが同期していない場合、一方向ジッターとパケット損失の場合はデータが返りますが、UDP ジッター動作による一方向遅延測定の場合は 0 の値が返ります。

- IP SLA アプリケーションを構成する前に、 **show ip sla application** コマンドを使用して、ソフトウェア イメージで目的の動作タイプがサポートされていることを確認してください。

UDP ジッター動作に関する注意事項と制約事項

- キーワードが付いている**show**コマンド**internal**はサポートされていません。
- 一方向遅延（レイテンシ）測定では、マイクロ秒単位の測定はサポートされていません。ミリ秒などの他の測定単位はサポートされています。

IP SLA パケットの CoPP の構成

IP SLA 動作を大規模なスケールで使用する場合、IP SLA パケットのパススルーを許可する特定の CoPP 構成が必要になる場合があります。IP SLA ではユーザー定義の UDP ポートを使用するため、コントロール プレーンへのすべての IP SLA パケットを許可する手段がありません。ただし、IP SLA が使用できる接続先/送信元ポートのそれぞれを指定することはできます。

IP SLA プローブ数の検証済みの拡張性に関する詳細については、*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*を参照してください。

以下に、IP SLA パケットのパススルーを許可する CoPP 構成例を示します。この例では、接続先ポートと送信元ポートが 6500 ～ 7000 の範囲であることを前提としています。この例では、「insert-before」が指定されていない場合、「class-default」の後に「copp-ipsla」が追加されます。

（注） 次の構成例は、プラットフォーム/ハードウェア タイプによって異なる場合があります。IP ACL および CoPP の設定の詳細については、『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』を参照してください。

```
ip access-list acl-sla-allow
  10 remark ### ALLOW SLA control packets from 1.1.1.0/24
  20 permit udp 1.1.1.0/24 any eq 1967
  30 remark ### ALLOW SLA data packets from 1.1.1.0/24 using ports 6500-7000
  40 permit udp 1.1.1.0/24 any range 6500 7000

class-map type control-plane match-any copp-ipsla
  match access-group name acl-sla-allow

policy-map type control-plane Custom-copp-policy-strict
    class copp-ipsla insert-before Custom-copp-class-l2-default
    police cir 1500 kbps

control-plane
  service-policy input Custom-copp-policy-strict

switch# show policy-map interface control-plane | be copp-ipsla
    class-map copp-ipsla (match-any)
      match access-group name acl-sla-allow
```

```
  set cos 7
  police cir 1500 kbps , bc 32000 bytes
  module 1 :
    transmitted 0 bytes;
    dropped 0 bytes;

class-map Custom-copp-class-l2-default (match-any)
  match access-group name Custom-copp-acl-mac-undesirable
  set cos 0
  police cir 400 kbps , bc 32000 bytes
  module 1 :
    transmitted 0 bytes;
    dropped 0 bytes;

class-map class-default (match-any)
  set cos 0
  police cir 400 kbps , bc 32000 bytes
  module 1 :
    transmitted 122 bytes;
    dropped 0 bytes;
```

Netstack ポート範囲の一致

IP SLA は、ローカルのネットスタック ポート範囲内のポートのみを受け入れます。プローブの設定で使用される送信元ポートと接続先ポートは、SLA 送信側と SLA レスポンダでサポートされている netstack ポートと一致している必要があります。

以前のバージョンからバージョン 9.3(1) 以降のバージョンに ISSU を実行する場合は、SSH ポートなどのユーザー定義ポートの機能が次の表に記載されている範囲内にあることを確認してください。

表 *3 : ISSU* のポート範囲

バージョン	デフォルトのポート範囲
9.3(1)	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 63535） nat ポート範囲（63536 ～ 65535）
9.3(2)	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 63535） nat ポート範囲（63536 ～ 65535）
9.3(3) 以降	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 60535） nat ポート範囲（60536 - 65535）

show sockets local-port-range コマンドを使用すれば コマンドは、送信側/応答側のポート範囲を表示します。

以下は、netstack ポート範囲を表示する例です。

```
switch# show sockets local-port-range

Kstack local port range (15001 - 22002)
Netstack local port range (22003 - 65535)
```

送信元デバイスでの UDP ジッター動作の設定およびスケジューリング

ここでは、UDP ジッター動作を構成し、スケジュールする方法について説明します。

宛先デバイスでの IP SLA Responder の設定

この項では、接続先デバイスでレスポンダを設定する方法について説明します。

（注） Responder では、同じ送信元に対して固定ポートを設定しないでください。Responder が同じ送信元に対して固定ポートを設定すると、パケットが正常に（タイムアウトまたはパケット損失の問題が発生せずに）送信されたとしても、ジッター値はゼロになります。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla responder**
4. 次のいずれかを実行します。
   - **ip sla responder**

     Example: `switch(config)# ip sla responder`
   - **ip sla responder udp-echo ipaddress** *ip-address* **port** *port*

     Example: `switch(config)# ip sla responder udp-echo ipaddress 172.29.139.132 port 5000`
5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla responder** 例： `switch(config)# feature sla responder`	IP SLA のレスポンダ機能を有効にします。
ステップ 4	次のいずれかを実行します。 • **ip sla responder** *Example:* `switch(config)# ip sla responder` • **ip sla responder udp-echo ipaddress** *ip-address* **port** *port* *Example:* `switch(config)# ip sla responder udp-echo ipaddress 172.29.139.132 port 5000`	- • （任意）送信元からの制御メッセージに応じて、Cisco デバイスにおけるレスポンダ機能を一時的に有効にします。 • （任意）送信元でプロトコル制御が無効である場合にのみ必須です。指定された IP アドレスおよびポートでレスポンダ機能を永続的に有効にします。 制御は、デフォルトでイネーブルになります。
ステップ 5	**exit** 例： `switch(config)# exit`	（任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

送信元デバイスでの基本的なUDPジッター動作の設定およびスケジューリング

ここでは、送信元デバイスでの基本 UDP ジッター動作を設定およびスケジュールする方法について説明します。

ヒント

- IP SLA 動作が実行せず、統計情報が生成されていない場合は、動作の設定に **verify-data** コマンドを追加して（IP SLA 構成モードで設定）、データ検証を有効にします。イネーブルになると、各動作の応答が破損していないかどうかがチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- IP SLA 動作に関する問題のトラブルシューティングを行うには、**debug ip sla sender trace** コマンドと **debug ip sla sender error** コマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla sender**
4. **ip sla** *operation-number*
5. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**sourceport** *port-number*] [**control** { **enable**| **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
6. **frequency** *seconds*
7. **exit**
8. **ip sla schedule** *operation-number* [**life** {*forever*| *seconds*}] [**start-time** {*hh:mm*[*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*}] [**ageout** *seconds*] [**recurring**]
9. **exit**
10. **show ip sla configuration** [*operation-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch# enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 4	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。

	コマンドまたはアクション	目的
ステップ 5	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**sourceport** *port-number*] [**control** { **enable**| **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] 例： `switch(config-ip-sla)# udp-jitter 172.29.139.134 5000`	IP SLA 動作を UDP ジッター動作として設定し、UDP ジッタ コンフィギュレーション サブモードを開始します。 送信元スイッチとターゲット スイッチの両方で IP SLA 制御プロトコルを無効にする場合のみ、**control disable** キーワードの組み合わせを使用します。
ステップ 6	**frequency** *seconds* 例： `switch(config-ip-sla-jitter)# frequency 30`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 7	**exit** 例： `switch(config-ip-sla-jitter)# exit`	UDP ジッタ コンフィギュレーション サブモードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 8	**ip sla schedule** *operation-number* [**life** {*forever*| *seconds*}] [**start-time** {*hh:mm*[*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*}] [**ageout** *seconds*] [**recurring**] 例： `switch(config)# ip sla schedule 5 start-time now life forever`	個々の IP SLA 動作のスケジューリング パラメータを設定します。
ステップ 9	**exit** 例： `switch(config)# exit`	（任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 10	**show ip sla configuration** [*operation-number*] 例： `switch# show ip sla configuration 10`	（任意）すべての IP SLA 動作または指定した IP SLA 動作に関する設定値を、すべてのデフォルト値を含めて表示します。

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

IP SLA 動作の結果を表示し、内容を確認するには、**show ip sla statistics** コマンドを使用します。サービス レベル契約の基準に対応するフィールドの出力を確認すると、サービス メトリックが許容範囲内であるかどうかを判断する役に立ちます。

追加特性を指定したUDPジッター動作の設定およびスケジューリング

ここでは、追加特性を使用して UDP ジッター動作を設定し、スケジュールする方法について説明します。

- UDP ジッター動作には大量のデータが含まれるので、以下のコマンド群は UDP ジッター動作ではサポートされず、そのため IP SLA UDP ジッター動作では IP SLA 履歴機能（統計情報の履歴バケット）はサポートされません：**history buckets-kept** 、**history filter**、**historylives-kept**、**samples-of-history-kept**、および **show ip sla history**。
- UDP ジッター動作の統計情報保存時間は、IP SLA で使用される MIB（CISCO-RTTMON-MIB）によって 2 時間に制限されます。**history hours-of-statistics** を使用してより大きな値を構成する*hours* グローバル構成を使用しても、保持される期間が 2 時間を超えることはありません。ただし、Data Collection MIB を使用して動作の履歴データを収集することはできます。詳細については、CISCO-DATA-COLLECTION-MIB（http://www.cisco.com/go/mibs）を参照してください。

ヒント

- IP SLA 動作が実行されておらず、統計を生成していない場合は、 **verify-data** コマンドを動作の構成に追加して（IP SLA 構成モードで設定）、データ検証を有効にします。イネーブルになると、各動作の応答が破損していないかどうかがチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- **debug ip sla sender trace** コマンドを使用し、 および **debug ip sla sender error** IP SLA 動作に関する問題をトラブルシューティングするコマンドです。

始める前に

送信元デバイスで UDP ジッタ動作を設定する前に、ターゲット デバイス（動作ターゲット）で IP SLA Responder をイネーブルにしておく必要があります。IP SLA Responder を使用できるのは、Cisco NX-OS ソフトウェアベースのデバイスだけです。Responder をイネーブルにするために、「接続先デバイスでの IP SLA Responder の設定」の項の作業を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla sender**
4. **ip sla** *operation-number*
5. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets***number-of-packets*] [**interval** *interpacket-interval*]
6. **history distributions-of-statistics-kept** *size*
7. **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*]
8. **frequency** *seconds*
9. **history hours-of-statistics-kept** *hours*
10. **owner** *owner-id*

11. **request-data-size** *bytes*
12. **history statistics-distribution-interval** *milliseconds*
13. **tag** *text*
14. **threshold** *milliseconds*
15. **timeout** *milliseconds*
16. **tos** *number*
17. **verify-data**
18. **vrf** *vrf-name*
19. **exit**
20. **ip sla schedule** *operation-number* [**life** {**forever**| *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after***hh*:*mm*:*ss*}] [**ageout***seconds*] [**recurring**]
21. **exit**
22. **show ip sla configuration** [*operation-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `Switch# configure terminal`	グローバル設定モードを開始します。
ステップ 3	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 4	**ip sla** *operation-number* 例： `Switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 5	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets***number-of-packets*] [**interval** *interpacket-interval*] 例： `Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000`	IP SLA 動作を UDP ジッター動作として設定し、UDP ジッタ コンフィギュレーション サブモードを開始します。 • **control disable** コマンドを使用し、キーワードの組み合わせは、送信元スイッチとターゲットスイッチの両方で IP SLA 制御プロトコルを無効にする場合のみ、使用してください。

	コマンドまたはアクション	目的
ステップ 6	**history distributions-of-statistics-kept** *size* 例： `Switch(config-ip-sla-jitter)# history distributions-of-statistics-kept 5`	（任意）IP SLA 動作中にホップ単位で保持する統計情報の配信数を設定します。
ステップ 7	**history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*] 例： `Switch(config-ip-sla-jitter)# history enhanced interval 900 buckets 100`	（任意）IP SLA 動作に対する拡張履歴収集をイネーブルにします。
ステップ 8	**frequency** *seconds* 例： `Switch(config-ip-sla-jitter)# frequency 30`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 9	**history hours-of-statistics-kept** *hours* 例： `Switch(config-ip-sla-jitter)# history hours-of-statistics-kept 4`	（任意）IP SLA 動作の統計情報を保持する時間数を設定します。
ステップ 10	**owner** *owner-id* 例： `Switch(config-ip-sla-jitter)# owner admin`	（任意）IP SLA 動作の簡易ネットワーク管理プロトコル（SNMP）所有者を設定します。
ステップ 11	**request-data-size** *bytes* 例： `Switch(config-ip-sla-jitter)# request-data-size 64`	（任意）IP SLA 動作の要求パケットのペイロードにおけるプロトコル データ サイズを設定します。
ステップ 12	**history statistics-distribution-interval** *milliseconds* 例： `Switch(config-ip-sla-jitter)# history statistics-distribution-interval 10`	（任意）IP SLA 動作で維持する各統計情報の配信間隔を設定します。
ステップ 13	**tag** *text* 例： `Switch(config-ip-sla-jitter)# tag TelnetPollServer1`	（任意）IP SLA 動作のユーザー指定 ID を作成します。

	コマンドまたはアクション	目的
ステップ 14	**threshold** *milliseconds* 例： `Switch(config-ip-sla-jitter)# threshold 10000`	（任意）IP SLA 動作によって作成されるネットワーク モニタリング統計情報を計算するための上限しきい値を設定します。
ステップ 15	**timeout** *milliseconds* 例： `Switch(config-ip-sla-jitter)# timeout 10000`	（任意）IP SLA 動作がその要求パケットからの応答を待機する時間を設定します。
ステップ 16	**tos** *number* 例： `Switch(config-ip-sla-jitter)# tos 160`	（任意）IPv4 ネットワークに限り、IP SLA 動作の IPv4 ヘッダーの ToS バイトを定義します。
ステップ 17	**verify-data** 例： `Switch(config-ip-sla-jitter)# verify-data`	（任意）IP SLA 動作が各応答パケットに対してデータ破壊の有無をチェックするようにします。
ステップ 18	**vrf** *vrf-name* 例： `Switch(config-ip-sla-jitter)# vrf vpn-A`	（任意）IP SLA 動作を使用して、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）内をモニタリングできるようにします。
ステップ 19	**exit** 例： `Switch(config-ip-sla-jitter)# exit`	UDP ジッタ コンフィギュレーション サブモードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 20	**ip sla schedule** *operation-number* [**life** {**forever**| *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after***hh*:*mm*:*ss*}] [**ageout***seconds*] [**recurring**] 例： `Switch(config)# ip sla schedule 5 start-time now life forever`	個々の IP SLA 動作のスケジューリング パラメータを設定します。
ステップ 21	**exit** 例： `Switch(config)# exit`	（任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 22	**show ip sla configuration** [*operation-number*] 例： `Switch# show ip sla configuration 10`	（任意）すべての IP SLA 動作または指定した IP SLA 動作に関する設定値を、すべてのデフォルト値を含めて表示します。

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

IP SLA 動作の結果を表示し、内容を確認するには、**show ip sla statistics** コマンドを使用します。 を実行する前に、ユーザ名がフィギュレーション ファイルに指定されていることを確認してください。サービス レベル契約の基準に対応するフィールドの出力を確認すると、サービス メトリックが許容範囲内であるかどうかを判断する役に立ちます。

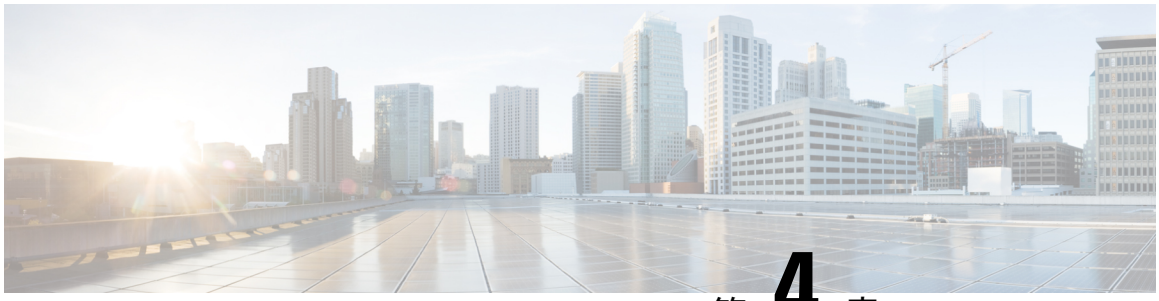
UDP ジッター動作の構成例

以下に、動作2が最初の動作の5秒後に開始されるUDPジッター動作として構成されている、2 つの動作を示します。どちらの動作も無期限に実行されます。

```
feature sla sender
ip sla 1
 udp-jitter 20.0.10.3 65051 num-packets 20
 request-data-size 160
 tos 128
 frequency 30
ip sla schedule 1 start-time after 00:05:00
ip sla 2
 udp-jitter 20.0.10.3 65052 num-packets 20 interval 10
 request-data-size 20
 tos 64
 frequency 30
ip sla schedule 2 start-time after 00:05:05
```

ターゲット（宛先）デバイスの設定は、次のとおりです。

```
feature sla responder
ip sla responder
```



第 4 章

VoIP 用の IP SLA UDP ジッター動作の設定

この章では、IP サービス レベル契約（SLA）ユーザー データグラム プロトコル（UDP）ジッター動作を構成してネットワーク内の Voice over IP（VoIP）品質レベルを予防的にモニタし、IPv4 または IPv6 ネットワーク内のユーザーに VoIP 品質レベルを保証できるようにする方法について説明します。IP SLA VoIP UDP ジッター動作は、共通のコーデックを使用して VoIP トラフィックを正確にシミュレートし、平均オピニオン評点（MOS）および Calculated Planning and Improvement Factor（ICPIF）などの一貫した音声品質スコアを計算します。

（注） このマニュアルでは、音声という用語はインターネット テレフォニー アプリケーションを示します。「Voice over IP」という用語には、IP ネットワーク経由のマルチメディア（音声とビデオの両方）の伝送が含まれることもあります。

この章は、次の項で構成されています。

VoIP 用の IP SLA UDP ジッター動作に関する注意事項と制約事項

- キーワードが付いている**show**コマンド**internal**はサポートされていません。

- この機能は、UDP トラフィックを使用して適切な Voice over IP スコアを生成します。Real-Time Transport Protocol（RTP）はサポートされていません。
- この機能で算出される Calculated Planning Impairment Factor（ICPIF）値および MOS 値は IP SLA 内では一貫していますが、相対的に比較するために生成された予想値に過ぎません。これらの値は、他の方法で測定された値とは異なる可能性があります。
- 任意の方法で測定されたカスタマーオピニオンの予測値（E-Model 伝送評価係数 R や算出された平均オピニオン評点に対して示された値など）は、伝送計画および分析のみを目的として生成された値です。実際のカスタマー オピニオンを反映する値ではありません。
- 一方向遅延（レイテンシ）測定では、マイクロ秒単位の測定はサポートされていません。ミリ秒などの他の測定単位はサポートされています。

計算された予定減損因子

ICPIF は、式 Icpif = Itot - A の一部として、1996 年版の ITU-T 勧告 G.113『Transmission impairments』で最初に開発されました。ICPIF とは、「計算された予定減損因子」（calculated planning impairment factor）の略です。ICPIF は、比較および計画用に、ネットワークに生じた音声品質に対する主な劣化の定量化を試みます。

ICPIF は、測定された劣化係数の合計（総劣化、つまり Itot）からユーザー定義のアクセス アドバンテージ係数（A）を引いたものです。アクセス アドバンテージ係数（A）は、通話方法（携帯電話からの通話対固定電話からの通話など）に基づいた、ユーザーの期待を表す値です。この式を拡張した、完全な式は次のようになります。

Icpif = Io + Iq + Idte + Idd + Ie - A

値は次のとおりです。

- *Io* は、最適ではないラウドネス定格が原因の劣化を表します。
- *Iq* は、PCM の量子化歪みが原因の劣化を表します。
- *Idte* は、送話者エコーによる劣化を表します。
- *Idd* は、一方向の伝送の時間（一方向遅延）により発生した劣化を表します。
- *Ie* は、通話に使用されたコーデック タイプ、パケット損失など装置の影響が原因の劣化を表します。
- *A* は、アクセスの容易性の代償としてユーザーがある程度の劣化を許容するという事実による補正である、アクセスアドバンテージ係数（ユーザー期待係数とも呼ばれます）を表します。

ICPIF の値は、通常、5（非常に軽い障害）から 55（非常に重い障害）の範囲で表されます。20 未満の ICPIF 値は、通常、「適切」と見なされます。ICPIF 値の目的は音声品質の客観的測定ですが、この値は、劣化の組み合わせの主観的影響を予測するためにも使用されます。

G.113（1996 年 2 月）に記載された、主観的品質判定に対応することが期待されるサンプル ICPIF 値を、次の図に示します。

ICPIF の上限	音声通信の品質
5	きわめて良好
10	良好
20	適切
30	限定された状況で許容可
45	きわめて限定された状況で許容可
55	ユーザーが強い不満を示す可能性が高い（苦情、ネットワーク オペレータの変更）

ICPIF の詳細については、1996 年版の G.113 の仕様を参照してください。

（注） 最新版の ITU-T G.113 勧告（2001 年）には、ICPIF モデルについての記載はありません。代わりに、現在は G.107 に記述されているように、ITU-T G.107 の E-model で使用される『劣化係数法』が推奨されています。量子化歪み単位を使用していた初期の方法は、現在では推奨されません」と記述されています。完全な E-Model（ITU-T 伝送評価モデルとも呼ばれます）は、*R = Ro - Is - Id - Ie + A* として表現され、劣化係数の定義の改善により、コール品質のより正確な測定の可能性を提供します（詳細については、G.107、2003 年版を参照してください）。ICPIF と E-Model は劣化に関する用語を共有していますが、これら 2 つのモデルは異なります。IP SLA VoIP UDP 動作機能では、ICPIF、伝送評価係数 R、および MOS 値の間で観測された対応関係が活用されていますが、E-Model はサポートされていません。

平均オピニオン評点

伝送される音声の品質は、聞き手の主観的な反応です。Voice over IP の伝送に使用する各コーデックは特定のレベルの品質を提供します。特定のコーデックによってもたらされる音質の測定に使用される共通のベンチマークは、平均オピニオン評点（MOS）です。MOS では、幅広い聞き手が、特定のコーデックを使用して送信された音声サンプルの品質を 1（貧弱）～5（優良）で判定します。オピニオン評点は平均化されて、各サンプルの平均が算出されます。

次の表に、各値に対する MOS 評点および対応する品質の説明を示します。

表 *4: MOS* 評価

スコア	品質	品質劣化の説明
5	優良	ほとんど感じられない
4	良	わずかに感じられるが、気にならない
3	可	感じられ、やや気になる
2	貧弱	気になるが、不快ではない
1	不可	非常に気になり、不快である

コーデックおよび他の伝送劣化に関する MOS 評点がよく知られているため、測定された劣化に基づいて MOS の予測値を算出し、表示できます。この予測値は、客観的 MOS または主観的 MOS 値と区別するために、ITU によって Mean Opinion Score; Conversational Quality, Estimated（MOS-CQE）と指定されました（詳細は、P.800.1 を参照）。

IP SLA を使用した音声パフォーマンスのモニタリング

IP ネットワーク上で音声品質およびビデオ品質を測定する際に重要なメトリックの 1 つはジッタです。ジッターは、受信パケット間の遅延における変動（パケット間の遅延のばらつき）の影響を示します。ジッタは、通話者の音声パターンに不均等なずれを生じさせて、音声品質に影響を与えます。IP ネットワーク上での音声伝送およびビデオ伝送に関するその他の重要なパフォーマンス パラメータには、遅延やパケット損失が挙げられます。IP SLA を使用してこれらのパラメータをシミュレートし、測定することで、ネットワークがユーザーとのサービス レベル契約を満たしている、または超えているかを確認できます。

IP SLA は、送信元デバイスから特定の接続先（動作ターゲットと呼ばれます）にネットワーク経由で UDP プローブ パケットを送信することにより、UDP ジッター動作を提供します。この合成トラフィックは、接続のジッタ量、ラウンドトリップ時間、方向別パケット損失、および一方向遅延を記録するために使用されます「合成トラフィック」という用語は、ネットワーク トラフィックがシミュレートされていることを示します。つまり、トラフィックは、IP SLA によって生成されます。収集された統計情報の形式でのデータは、ユーザー定義による期間内の複数のテストに対して表示できます。たとえば、1 日の異なる時間の、または週の経過に伴うネットワークのパフォーマンスを確認できます。ジッタープローブでは、受信側での遅延を最小にするため、IP SLA Responder を使用できます。

IP SLA VoIP UDP ジッター動作は、UDP ジッター動作によって既に収集されているメトリックに加えて、動作によって収集されたデータに MOS スコアおよび ICPIF スコアを返す機能を追加することによって標準的な UDP ジッター動作を変更します。この VoIP 固有の実装により、VoIP ネットワークのパフォーマンスを判断することができます。

IP SLA でのコーデックのシミュレーション

IP SLA VoIP UDP ジッター動作は、指定された頻度 f で、指定された送信元スイッチから指定されたターゲット スイッチに、サイズ s の n 個の UDP パケットを t ミリ秒間隔で送信して統計情報を計算します。ターゲット スイッチは、プローブ動作を処理するために、IP SLA Responder を実行している必要があります。

MOS スコアと ICPIF スコアを生成するには、VoIP UDP ジッタ動作を設定するときに、接続に使用するコーデック タイプを指定します。動作に構成したコーデック タイプに基づいて、パケット数（n）、各ペイロードのサイズ（s）、パケット間間隔（t）、および動作の頻度（f）がデフォルト値に自動構成されますただし、必要な場合は、**udp-jitter** コマンドの構文でこれらのパラメータを手動で設定することもできます。 を実行する前に、ユーザ名がフィギュレーション ファイルに指定されていることを確認してください。

次の表に、コーデックによる動作に構成されるデフォルト パラメータを示します。

表 *5:* デフォルトの *VoIP UDP* ジッター動作パラメータ（コーデック タイプ別）

コーデック	デフォルトの要求サイズ（パケットペイロード）（**s**）	デフォルトのパケット間間隔（**t**）	デフォルトのパケット数（**n**）	プローブ動作の頻度（**f**）
G.711 mu-Law（g711ulaw）	160 + 12 RTP バイト	20 ms	1000	1 分に 1 回
G.711 A-Law（g711alaw）	160 + 12 RTP バイト	20 ms	1000	1 分に 1 回
G.729A（g729a）	20 + 12 RTP バイト	20 ms	1000	1 分に 1 回

たとえば、g711ulaw コーデックの特性を使用する VoIP UDP ジッター動作を設定した場合、プローブ動作はデフォルトで 1 分に 1 回（**f**）送信されます。各プローブ動作は 1000 パケット（**n**）で構成され、各パケットは 180 バイトの合成データ（**s**）を含み、20 ミリ秒間隔（**t**）で送信されます。

IP SLA ICPIF 値

Cisco NX-OS ソフトウェアを使用する際の ICPIF 値の計算は、主として音声品質を損なう 2 つの主要因（遅延パケットと損失パケット）に基づいています。パケット遅延とパケット損失は IP SLA で測定できます。したがって、ICPIF 式（*Icpif=Io+Iq+Idte+Idd+Ie-A*）は、*Io*、*Iq*、および *Idte* の値がゼロであると想定することによって簡素化され、次のようになります。

総劣化係数（*Icpif*） = 遅延劣化係数*r*（*Idd*） + 機器劣化係数（*Ie*） - 期待/アドバンテージ係数（*A*）

ICPIF 値は、遅延パケットの測定値に基づいた遅延劣化係数と、損失パケットの測定値に基づいた機器劣化係数を加算して算出されます。ネットワーク内で測定されたこの総劣化の合計値から劣化変数（期待係数）を引くと、ICPIF になります。

Cisco ゲートウェイは、受信した VoIP データ ストリームの ICPIF の計算には、この式を使用します。

遅延劣化係数

遅延劣化係数（*Idd*）は、2 つの値に基づいた数値です。1 つの値は、固定値です。（ITU 規格で規定された）コーデック遅延、先読み遅延、およびデジタル信号処理（DSP）遅延の固定値を使用して算出されます。2 番めの値は、変数です。測定された一方向遅延（ラウンドトリップ時間測定値を 2 で割った値）に基づいています。一方向遅延値は、G.107（2002 年版）の分析式に基づいたマッピング テーブルを使用して数値にマップされます。

次の表に、IP SLA によって測定された一方向遅延と遅延劣化係数値の対応関係の例を示します。

表 *6*：一方向遅延と *ICPIF* 遅延劣化係数の対応関係の例

一方向遅延（ミリ秒）	遅延劣化係数
50	1
100	2
150	4
200	7

機器劣化係数

機器劣化係数（*Ie*）は、測定されたパケット損失量に基づいた数値です。測定されたパケット損失量は総送信パケット数の割合として表され、コーデックによって定義される機器劣化係数に対応します。

次の表に、IP SLA によって測定されたパケット損失と機器劣化係数値（相互に対応）との間の対応関係の例を示します。

表 *7*：測定されたパケット損失と *ICPIF* 機器劣化の対応関係の例

パケット損失（送信済みパケットの総数のパーセント）	PCM（G.711）コーデックの機器劣化値	CS-ACELP（G.729A）コーデックの機器劣化値
2 %	12	20
4 %	22	30
6 %	28	38
8 %	32	42

期待係数

アドバンテージ係数（*A*）とも呼ばれる期待係数は、ユーザーがアクセスの容易性の代償としてある程度の品質の劣化を許容する可能性があるという予測を表します。たとえば、到達困難な場所にいる携帯電話ユーザーは、接続品質が従来の固定電話接続ほど良好ではないことを予測している可能性があります。この変数は、向上したアクセスの利便性と音声品質の低下の釣り合いを保つことを目的としているので、アドバンテージ係数（アクセスアドバンテージ係数の略）とも呼ばれます。

次の表は ITU-T 勧告G.113 を改良したもので、*A* の暫定最大値のセットを、提供されるサービスごとに定義しています。

表 *8*: アドバンテージ係数の推奨最大値

通信サービス	アドバンテージ/期待係数： A の最大値
従来の有線（固定電話）	0
建物内のモビリティ（セルラー接続）	5
地域内または車内のモビリティ	10
到達困難な場所へのアクセス（たとえば、マルチホップ衛星接続を介したアクセスなど）	20

これらの値は推奨値に過ぎません。意味のあるものにするには、係数（*A*）および特定のアプリケーションで選択したその値を一貫して、採用するすべてのプランニングモデルで使用する必要があります。ただし、上の表の値は、*A* の絶対的な上限と見なす必要があります。

IP SLA VoIP UDP ジッター動作のデフォルトのアドバンテージ係数は常に 0 です。

IP SLA MOS 値

IP SLA は、ICPIF 値と MOS 値との測定された対応関係を使用して MOS 値を予測します。

（注） 略語 MOS は MOSCQE（平均オピニオン評点、会話品質推定値 - Mean Opinion Score; Conversational Quality, Estimated）を表します。

G.107（2003 年 3 月）で定義された E-Model は、伝送パラメータが原因の劣化（損失、遅延など）を組み合わせて 1 つの評価、つまり伝送評価係数 R（R 係数）を算出することによって、平均的な聞き手が感じる主観的な品質を予測します。0（最低）～100（最高）で表されるこの評価は、MOS などユーザーの主観的な反応を予測するために使用されます。具体的には、MOS は R 係数から変換式を使用して算出できます。逆に言うと、この式を逆変換式に修正して使用すれば、MOS 値から R 係数を算出できます。

ICPIF 値と R 係数との間にも関係があります。IP SLA は、ICPIF スコアから算出された R 係数の予測値から適切な MOS スコアの概算値を算出して、この対応関係を利用します。

次の表に、対応する ICPIF 値に対して生成される MOS 値を示します。

表 *9: MOS* 値に対する *ICPIF* 値の対応関係

ICPIF の範囲	MOS	品質のカテゴリ
0 ～ 3	5	最良
4 ～ 13	4	高
14 ～ 23	3	中
24 ～ 33	2	小さい
34 ～ 43	1	きわめて小さい

IP SLA は、MOS 予測値を常に 1 ～ 5 で表します。5 が最高品質です。MOS 値が 0（ゼロ）の場合は、その動作に対して MOS データを生成できなかったことを示します。

IP SLA VoIP UDP ジッター動作の設定およびスケジューリング

（注）

- 現時点では、IP SLA は次の音声コーデック（圧縮法）のみをサポートします。
  - G.711 A Law（g711alaw: 64 kbps PCM 圧縮法）
  - G.711 mu Law（g711ulaw: 64 kbps PCM 圧縮法）
  - G.729A（g729a: 8 kbps CS-ACELP 圧縮法）
- 次のコマンドは UDP ジッタ コンフィギュレーション モードでは使用できますが、UDP ジッタ（コーデック）動作では使用できません。
  - **history distributions-of-statistics-kept**
  - **history statistics-distribution-interval**
  - **request-data-size**
- コーデック タイプを指定すると、**codec-interval**、**codec-size**、および **codec-numpacket** の各オプションに適切なデフォルト値が設定されます。デフォルト値よりも優先させる特別な理由（異なるコーデックの概算など）がある場合を除き、間隔、サイズ、およびパケット数の各オプションの値を指定しないでください。
- この項で説明している **show ip sla configuration** コマンドは、保持される統計分散バケット数と統計分散間隔（マイクロ秒）を表示しますが、これらの値はジッター（コーデック）動作には適用されません。

ヒント

- IP SLA 動作が実行されておらず、統計を生成していない場合は、 **verify-data** コマンドを動作の構成に追加して（IP SLA 構成モードで設定）、データ検証を有効にします。イネーブルになると、各動作の応答が破損していないかどうかがチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- **debug ip sla trace** コマンドを使用し、 および **debug ip sla error** コマンドは、IP SLA 動作に関する問題のトラブルシューティングを行うためのコマンドです。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla sender**
4. **ip sla** *operation-number*
5. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* **codec** *codec-type* [**codec-numpackets** *number-of-packets*] [**codec-size** *number-of-bytes*] [**codec-interval** *milliseconds*] [**advantage-factor** *value*] [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}]
6. **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*]
7. **frequency** *seconds*
8. **history hours-of-statistics-kept** *hours*
9. **owner** *owner-id*
10. **tag** *text*
11. **threshold** *microseconds*
12. **timeout** *microseconds*
13. **tos** *number*
14. **verify-data**
15. **vrf** *vrf-name*
16. **exit**
17. **ip sla schedule** *operation-number* [**life** {**forever**| *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *seconds*] [**recurring**]
18. **exit**
19. **show ip sla configuration** [*operation-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたら、パスワードを入力します。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 4	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 5	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* **codec** *codec-type* [**codec-numpackets** *number-of-packets*] [**codec-size** *number-of-bytes*] [**codec-interval** *milliseconds*] [**advantage-factor** *value*] [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] 例： `switch(config-ip-sla)# udp-jitter 209.165.200.225 16384 codec g711alaw advantage-factor 10`	遅延、ジッタ、およびパケット損失の統計情報に加えて、VoIP スコアを生成するジッタ（コーデック）動作としてこの動作を設定します。
ステップ 6	**history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*] 例： `switch(config-ip-sla-jitter)# history enhanced interval 900 buckets 100`	（任意）IP SLA 動作に対する拡張履歴収集をイネーブルにします。
ステップ 7	**frequency** *seconds* 例： `switch(config-ip-sla-jitter)# frequency 30`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 8	**history hours-of-statistics-kept** *hours* 例： `switch(config-ip-sla-jitter)# history hours-of-statistics-kept 4`	（任意）IP SLA 動作の統計情報を保持する時間数を設定します。
ステップ 9	**owner** *owner-id* 例：	（任意）IP SLA 動作の簡易ネットワーク管理プロトコル（SNMP）所有者を設定します。

	コマンドまたはアクション	目的
	`switch(config-ip-sla-jitter)# owner admin`	
ステップ 10	**tag** *text* 例： `switch(config-ip-sla-jitter)# tag TelnetPollServer1`	（任意）IP SLA 動作のユーザー指定 ID を作成します。
ステップ 11	**threshold** *microseconds* 例： `switch(config-ip-sla-jitter)# threshold 10000`	（任意）IP SLA 動作によって作成されるネットワーク モニタリング統計情報を計算するための上限しきい値を設定します。
ステップ 12	**timeout** *microseconds* 例： `switch(config-ip-sla-jitter)# timeout 10000`	（任意）IP SLA 動作がその要求パケットからの応答を待機する時間を設定します。
ステップ 13	**tos** *number* 例： `switch(config-ip-sla-jitter)# tos 160`	（任意）IPv4 ネットワークに限り、IP SLA 動作の IPv4 ヘッダーの ToS バイトを定義します。
ステップ 14	**verify-data** 例： `switch(config-ip-sla-jitter)# verify-data`	（任意）IP SLA 動作が各応答パケットに対してデータ破壊の有無をチェックするようにします。
ステップ 15	**vrf** *vrf-name* 例： `switch(config-ip-sla-jitter)# vrf vpn-A`	（任意）IP SLA 動作を使用して、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）内をモニタリングできるようにします。
ステップ 16	**exit** 例： `switch(config-ip-sla-jitter)# exit`	UDP ジッタ コンフィギュレーション サブモードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 17	**ip sla schedule** *operation-number* [**life** {**forever**| *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *seconds*] [**recurring**] 例： `switch(config)# ip sla schedule 5 start-time now life forever`	個々の IP SLA 動作のスケジューリング パラメータを設定します。

	コマンドまたはアクション	目的
ステップ 18	**exit** 例： `switch(config)# exit`	（任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 19	**show ip sla configuration** [*operation-number*] 例： `switch# show ip sla configuration 10`	（任意）すべての IP SLA 動作または指定した IP SLA 動作に関する設定値を、すべてのデフォルト値を含めて表示します。

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

IP SLA 動作の結果を表示し、内容を解釈するには、**show ip sla statistics** を実行する前に、ユーザ名がフィギュレーション ファイルに指定されていることを確認してください。サービス レベル契約の基準に対応するフィールドの出力を確認すると、サービスメトリックが許容範囲内であるかどうかを判断する役に立ちます。

IP SLA VoIP UDP 動作の設定例

次の例では、IP SLA Responder が 101.101.101.1 のデバイスで有効であることを前提とします。

```
switch# conf terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# feature sla sender
switch(config)# ip sla 10
switch(config-ip-sla)# udp-jitter 101.101.101.1 16384 codec g711alaw advantage-factor 2
switch(config-ip-sla-jitter)# owner admin_bofh
switch(config-ip-sla-jitter)# precision microseconds
switch(config-ip-sla-jitter)# exit
switch(config)# ip sla schedule 10 start-time now
switch(config)# exit
switch# show ip sla config 10
IP SLAs Infrastructure Engine-III
Entry number: 10
Owner: admin_bofh
Tag:
Operation timeout (milliseconds): 5000
Type of operation to perform: udp-jitter
Target address/Source address: 101.101.101.1/0.0.0.0
Target port/Source port: 16384/0
Type Of Service parameter: 0x0
Codec type: g711alaw
Codec Number Of Packets: 1000
Codec Packet Size: 172
Codec Interval (milliseconds): 20
Advantage Factor: 2
Verify data: No
Operation Stats Precision : microseconds
```

```
Operation Packet Priority : normal
NTP Sync Tolerance : 0 percent
Vrf Name: default
Control Packets: enabled
Schedule:
   Operation frequency (seconds): 60  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): 3600
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
   Number of statistic hours kept: 2
   Number of statistic distribution buckets kept: 1
   Statistic distribution interval (microseconds): 20

switch#

switch# show running-config | begin "ip sla 10"
ip sla 10
  udp-jitter 101.101.101.1 16384 codec g711alaw advantage-factor 2
    precision microseconds
    owner admin_bofh
ip sla schedule 10 start-time now
no logging console
.
.
.
switch# show ip sla configuration 10
Entry number: 10
Owner: admin_bofh
Tag:
Type of operation to perform: jitter
Target address: 101.101.101.1
Source address: 0.0.0.0
Target port: 16384
Source port: 0
Operation timeout (milliseconds): 5000
Codec Type: g711alaw
Codec Number Of Packets: 1000
Codec Packet Size: 172
Codec Interval (milliseconds): 20
Advantage Factor: 2
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Operation frequency (seconds): 60
Next Scheduled Start Time: Start Time already passed
Life (seconds): 3600
Entry Ageout (seconds): never
Status of entry (SNMP RowStatus): Active
Connection loss reaction enabled: No
Timeout reaction enabled: No
Verify error enabled: No
Threshold reaction type: Never
Threshold (milliseconds): 5000
Threshold Falling (milliseconds): 3000
Threshold Count: 5
Threshold Count2: 5
Reaction Type: None
```

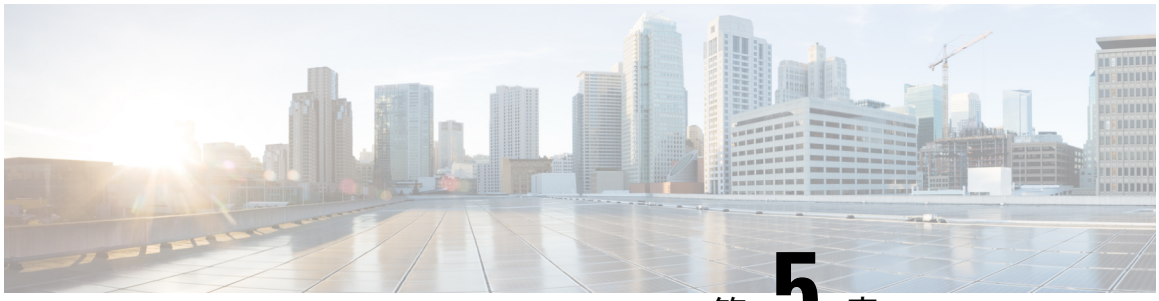
```
Number of statistic hours kept: 2
Number of statistic distribution buckets kept: 1
Statistic distribution interval (microseconds): 20
Enhanced History:
```

コーデック タイプがジッター動作用に構成されている場合、標準ジッターの「要求サイズ（ARR データ部）（Request size（ARR data portion））」、「パケット数（Number of packets）」、および「間隔（マイクロ秒）（Interval (microseconds)）」のパラメータは、**show ip sla** 構成コマンドの出力には表示されません。代わりに、「コーデック パケット サイズ（Codec Packet Size）」、「コーデック パケット数（Codec Number of Packets）」、および「コーデック間隔（マイクロ秒）（Codec Interval（microseconds））」が表示されます。

IP SLA VoIP UDP 動作統計情報の出力の設定例

以下に、ジッター（コーデック）動作の音声スコア（ICPIF 値と MOS 値）を表示する例を示します。

```
switch# show ip sla st
IPSLAs Latest Operation Statistics
IPSLA operation id: 1
Type of operation: udp-jitter
        Latest RTT: 11999 microseconds
Latest operation start time: 02:39:33 UTC Sat May 05 2012
Latest operation return code: OK
Latest operation NTP sync state: NO_SYNC
RTT Values:
        Number Of RTT: 10
RTT Min/Avg/Max: 9000/11999/17000 microseconds
Latency one-way time:
        Number of Latency one-way Samples: 0
        Source to Destination Latency one way Min/Avg/Max: 0/0/0 microseconds
        Destination to Source Latency one way Min/Avg/Max: 0/0/0 microseconds
Jitter Time:
        Number of SD Jitter Samples: 9
        Number of DS Jitter Samples: 9
        Source to Destination Jitter Min/Avg/Max: 0/223/2001 microseconds
        Destination to Source Jitter Min/Avg/Max: 0/2001/6001 microseconds
Packet Loss Values:
        Loss Source to Destination: 0
        Source to Destination Loss Periods Number: 0
```



第 5 章

IP SLA UDP エコー動作の設定

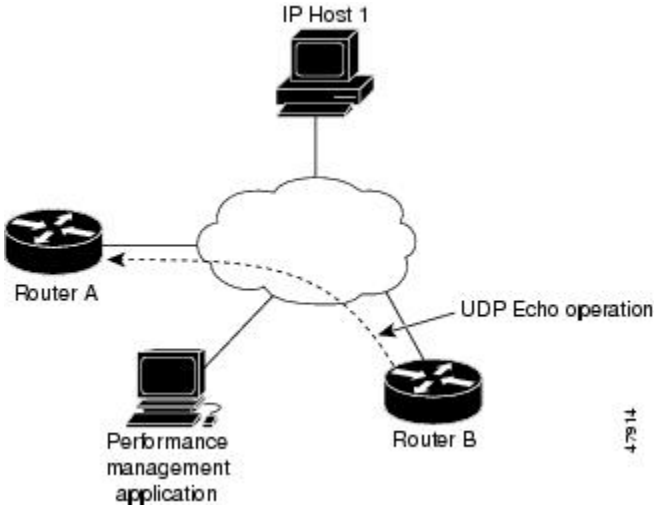
この章では、IP サービス レベル契約（SLA）ユーザ データグラム プロトコル（UDP）エコー動作を設定して、Cisco スイッチと IPv4 を使用するデバイスとの間のエンドツーエンド応答時間をモニタする方法について説明します。UDP エコーの精度は、接続先の Cisco スイッチで IP SLA Responder を使用することによって向上します。このモジュールでは、UDP エコー動作の結果を表示して分析し、UDP アプリケーションのパフォーマンスを測定する方法についても説明します。

この章は、次の項で構成されています。

UDP エコー動作

UDP エコー動作は、Cisco スイッチと IP を使用するデバイスとの間でエンドツーエンド応答時間を測定します。UDP は、多くの IP サービスで使用されるトランスポート層（レイヤ 4）インターネット プロトコルです。UDP エコーは応答時間を測定し、エンドツーエンドの接続をテストするために使用されます。

次の図では、スイッチ A が IP SLA Responder として設定され、スイッチ B が送信元 IP SLA デバイスとして設定されています。



スイッチ B から宛先スイッチ（スイッチ A）に UDP エコー要求メッセージを送信し、スイッチ A からの UDP エコー応答を受信するまでの時間を測定することで、応答時間（ラウンドトリップ時間）が算出されます。UDP エコーの精度は、スイッチ A（宛先の Cisco スイッチ）でレスポンダを使用することによって向上します。宛先スイッチが Cisco スイッチの場合、IP SLA Responder は指定した任意のポート番号に UDP データグラムを送信します。シスコ デバイスを使用する場合、UDP エコー動作における IP SLA Responder の使用は任意です。シスコ以外のデバイスに IP SLA Responder を設定することはできません。

ラウンド トリップ遅延時間を測定し、Cisco および Cisco 以外のデバイス両方への接続をテストすることによって、ビジネス クリティカルなアプリケーションに関連した問題のトラブルシューティングを行う際に、UDP エコー動作の結果が役立つことがあります。

UDP エコー動作に関する注意事項と制約事項

- キーワードが付いている**show**コマンド**internal**はサポートされていません。

IP SLA パケットの CoPP の構成

IP SLA 動作を大規模なスケールで使用する場合、IP SLA パケットのパススルーを許可する特定の CoPP 構成が必要になる場合があります。IP SLA ではユーザー定義の UDP ポートを使用するため、コントロール プレーンへのすべての IP SLA パケットを許可する手段がありません。ただし、IP SLA が使用できる接続先/送信元ポートのそれぞれを指定することはできます。

IP SLA プローブ数の検証済みの拡張性に関する詳細については、*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*を参照してください。

以下に、IP SLA パケットのパススルーを許可する CoPP 構成例を示します。この例では、接続先ポートと送信元ポートが 6500 ～ 7000 の範囲であることを前提としています。この例では、「insert-before」が指定されていない場合、「class-default」の後に「copp-ipsla」が追加されます。

（注） 次の構成例は、プラットフォーム/ハードウェア タイプによって異なる場合があります。IP ACL および CoPP の設定の詳細については、『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』を参照してください。

```
ip access-list acl-sla-allow
  10 remark ### ALLOW SLA control packets from 1.1.1.0/24
  20 permit udp 1.1.1.0/24 any eq 1967
  30 remark ### ALLOW SLA data packets from 1.1.1.0/24 using ports 6500-7000
  40 permit udp 1.1.1.0/24 any range 6500 7000

class-map type control-plane match-any copp-ipsla
  match access-group name acl-sla-allow

policy-map type control-plane Custom-copp-policy-strict
    class copp-ipsla insert-before Custom-copp-class-l2-default
    police cir 1500 kbps

control-plane
  service-policy input Custom-copp-policy-strict

switch# show policy-map interface control-plane | be copp-ipsla
    class-map copp-ipsla (match-any)
      match access-group name acl-sla-allow
      set cos 7
      police cir 1500 kbps , bc 32000 bytes
      module 1 :
        transmitted 0 bytes;
        dropped 0 bytes;

    class-map Custom-copp-class-l2-default (match-any)
      match access-group name Custom-copp-acl-mac-undesirable
      set cos 0
      police cir 400 kbps , bc 32000 bytes
      module 1 :
        transmitted 0 bytes;
        dropped 0 bytes;

    class-map class-default (match-any)
      set cos 0
      police cir 400 kbps , bc 32000 bytes
      module 1 :
        transmitted 122 bytes;
        dropped 0 bytes;
```

Netstack ポート範囲の一致

IP SLA は、ローカルのネットスタック ポート範囲内のポートのみを受け入れます。プローブの設定で使用される送信元ポートと接続先ポートは、SLA 送信側と SLA レスポンダでサポートされている netstack ポートと一致している必要があります。

以前のバージョンからバージョン 9.3(1) 以降のバージョンに ISSU を実行する場合は、SSH ポートなどのユーザー定義ポートの機能が次の表に記載されている範囲内にあることを確認してください。

表 *10 : ISSU* のポート範囲

バージョン	デフォルトのポート範囲
9.3(1)	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 63535） nat ポート範囲（63536 ～ 65535）
9.3(2)	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 63535） nat ポート範囲（63536 ～ 65535）
9.3(3) 以降	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 60535） nat ポート範囲（60536 - 65535）

show sockets local-port-range コマンドを使用すれば コマンドは、送信側/応答側のポート範囲を表示します。

以下は、netstack ポート範囲を表示する例です。

```
switch# show sockets local-port-range

Kstack local port range (15001 - 22002)
Netstack local port range (22003 – 65535)
```

宛先デバイスでの IP SLA Responder の設定

始める前に

IP SLA Responder を使用する場合は、応答側として使用するネットワーキング デバイスがシスコ デバイスであり、そのデバイスにネットワークを介して接続できることを確認します。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla responder**
4. 次のいずれかを実行します。
   - **ip sla responder**

例：

```
switch(config)# ip sla responder
```

- **ip sla responder udp-echo ipaddress** *ip-address* **port** *port*

  例：

  ```
  switch(config)# ip sla responder udp-echo ipaddress 172.29.139.132 port 5000
  ```

5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードをイネーブルにします プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla responder** 例： `switch(config)# feature sla responder`	IP SLA のレスポンダ機能を有効にします。

ステップ 4	次のいずれかを実行します。
	• **ip sla responder**
	例：
	`switch(config)# ip sla responder`
	• **ip sla responder udp-echo ipaddress** *ip-address* **port** *port*
	例：
	`switch(config)# ip sla responder udp-echo ipaddress 172.29.139.132 port 5000`
		-
		• 送信元からの制御メッセージに応じて、Cisco デバイスにおける IP SLA Responder 機能を一時的に有効にします。
		• 送信元でプロトコル制御が無効である場合にのみ必須です。このコマンドは、指定の IP アドレスおよびポートで IP SLA Responder 機能を永続的に有効にします。
		制御は、デフォルトでイネーブルになります。

ステップ 5	**exit** 例： `switch(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

送信元デバイスでの基本 UDP エコー動作の設定

ここでは、送信元での基本 UDP エコー動作を構成する方法について説明します。

（注） トラップを生成する目的、または別の動作を開始する目的で、IP SLA 動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

始める前に

IP SLA Responder を使用する場合は、このタスクを開始する前に「宛先デバイスでの IP SLA Responder の設定」の項を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla** *operation-number*
4. **udp-echo** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **sourceport** *port-number*] [**control** {**enable** | **disable**}]
5. （任意） **frequency** *seconds*
6. （任意） **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 4	**udp-echo** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **sourceport** *port-number*] [**control** {**enable** | **disable**}] 例： `switch(config-ip-sla)# udp-echo 172.29.139.134 5000`	UDP エコー動作を定義し、IP SLA UDP コンフィギュレーション モードを開始します。 送信元スイッチとターゲット スイッチの両方で IP SLA 制御プロトコルを無効にする場合のみ、control disable キーワードの組み合わせを使用します。
ステップ 5	（任意） **frequency** *seconds* 例：	指定した IP SLA 動作を繰り返す間隔を設定します。

	コマンドまたはアクション	目的
	`switch(config-ip-sla-udp)# frequency 30`	
ステップ 6	（任意） **end** 例： `switch(config-ip-sla-udp)# end`	特権 EXEC モードに戻ります。

送信元デバイスでのオプション パラメータを使用した UDP エコー動作の設定

ここでは、送信元デバイスでオプション パラメータを使用して UDP エコー動作を構成する方法について説明します。

（注） トラップを生成する目的、または別の動作を開始する目的で、IP SLA 動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

始める前に

この動作で IP SLA Responder を使用している場合、宛先デバイスで Responder を設定する必要があります。「接続先デバイスでの IP SLA Responder の構成」を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla** *operation-number*
4. **udp-echo** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **sourceport** *port-number*] [**control** {**enable** | **disable**}]
5. （任意） **history buckets-kept** *size*
6. （任意） **data-pattern** *hex-pattern*
7. （任意） **history distributions-of-statistics-kept** *size*
8. （任意） **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*]
9. （任意） **history filter** {**none** | **all** | **overThreshold** | **failures**}
10. （任意） **frequency** *seconds*
11. （任意） **history hours-of-statistics-kept** *hours*
12. （任意） **history lives-kept** *lives*
13. （任意） **owner** *owner-id*
14. （任意） **request-data-size** *bytes*
15. （任意） **history statistics-distribution-interval** *milliseconds*

16. （任意） **tag** *text*
17. （任意） **threshold** *milliseconds*
18. （任意） **timeout** *milliseconds*
19. （任意） **tos** *number*
20. （任意） **verify-data**
21. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 4	**udp-echo** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **sourceport** *port-number*] [**control** {**enable** | **disable**}] 例： `switch(config-ip-sla)# udp-echo 172.29.139.134 5000`	UDP エコー動作を定義し、IP SLA UDP コンフィギュレーション モードを開始します。 送信元スイッチとターゲット スイッチの両方で IP SLA 制御プロトコルを無効にする場合のみ、control disable キーワードの組み合わせを使用します。
ステップ 5	（任意） **history buckets-kept** *size* 例： `switch(config-ip-sla-udp)# history buckets-kept 25`	IP SLA 動作のライフタイム中に保持する履歴バケット数を設定します。
ステップ 6	（任意） **data-pattern** *hex-pattern* 例： `switch(config-ip-sla-udp)# data-pattern`	データ破損のテストのために IP SLA 動作のデータ パターンを指定します。
ステップ 7	（任意） **history distributions-of-statistics-kept** *size* 例： `switch(config-ip-sla-udp)# history distributionsof- statistics-kept 5`	IP SLA 動作中にホップ単位で保持する統計情報の配信数を設定します。
ステップ 8	（任意） **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*]	IP SLA 動作に対する拡張履歴収集を有効にします。

	コマンドまたはアクション	目的
	例： `switch(config-ip-sla-udp)# history enhanced interval 900 buckets 100`	
ステップ 9	（任意） **history filter** {**none** | **all** | **overThreshold** | **failures**} 例： `switch(config-ip-sla-udp)# history filter failures`	IP SLA 動作の履歴テーブルに格納する情報のタイプを定義します。
ステップ 10	（任意） **frequency** *seconds* 例： `switch(config-ip-sla-udp)# frequency 30`	指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 11	（任意） **history hours-of-statistics-kept** *hours* 例： `switch(config-ip-sla-udp)# history hours-ofstatistics- kept 4`	IP SLA 動作の統計情報を保持する時間数を設定します。
ステップ 12	（任意） **history lives-kept** *lives* 例： `switch(config-ip-sla-udp)# history lives-kept 5`	IP SLA 動作の履歴テーブルに格納するライフ数を設定します。
ステップ 13	（任意） **owner** *owner-id* 例： `switch(config-ip-sla-udp)# owner admin`	IP SLA 動作の簡易ネットワーク管理プロトコル（SNMP）所有者を設定します。
ステップ 14	（任意） **request-data-size** *bytes* 例： `switch(config-ip-sla-udp)# request-data-size 64`	IP SLA 動作の要求パケットのペイロードにおけるプロトコル データ サイズを設定します。
ステップ 15	（任意） **history statistics-distribution-interval** *milliseconds* 例： `switch(config-ip-sla-udp)# history statistics distribution- interval 10`	IP SLA 動作で維持する各統計情報の配信間隔を設定します。
ステップ 16	（任意） **tag** *text* 例： `switch(config-ip-sla-udp)# tag TelnetPollServer1`	IP SLA 動作のユーザー指定 ID を作成します。
ステップ 17	（任意） **threshold** *milliseconds* 例： `switch(config-ip-sla-udp)# threshold 10000`	IP SLA 動作によって作成されるネットワーク モニタリング統計情報を計算するための上限しきい値を設定します。

	コマンドまたはアクション	目的
ステップ 18	（任意） **timeout** *milliseconds* 例： `switch(config-ip-sla-udp)# timeout 10000`	IP SLA 動作がその要求パケットからの応答を待機する時間を設定します。
ステップ 19	（任意） **tos** *number* 例： `switch(config-ip-sla-jitter)# tos 160`	IPv4 ネットワークに限り、IP SLA 動作の IPv4 ヘッダーの ToS バイトを定義します。
ステップ 20	（任意） **verify-data** 例： `switch(config-ip-sla-udp)# verify-data`	IP SLA 動作が各応答パケットに対してデータ破壊の有無をチェックするようにします。
ステップ 21	**exit** 例： `switch(config-ip-sla-udp)# exit`	UDP コンフィギュレーション サブモードを終了し、グローバル コンフィギュレーション モードに戻ります。

IP SLA 動作のスケジューリング

ここでは、IP SLA 動作をスケジュールする方法について説明します。

始める前に

（注）
- スケジュールされるすべての IP SLA 動作がすでに設定されている必要があります。
- 複数動作グループでスケジュールされたすべての動作の頻度が同じでなければなりません。
- 複数動作グループに追加される 1 つ以上の動作 ID 番号のリストは、カンマ（,）を含めて最大 125 文字に制限されます。

ヒント
- IP SLA 動作が実行されておらず、統計を生成していない場合は、 **verify-data** コマンドを動作の構成に追加して（IP SLA 構成モードで設定）、データ検証を有効にします。イネーブルになると、各動作の応答が破損していないかどうかがチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- **debug ip sla trace** コマンドを使用し、および **debug ip sla error** コマンドは、IP SLA 動作に関する問題のトラブルシューティングを行うためのコマンドです。

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかを実行します。
   - **ip sla schedule** *operation-number* [**life forever** { | *seconds*}] [**starttime** {*hh : mm*[: *ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh : mm : ss*}] [**ageout** *seconds*] [**recurring**]

     例：

     ```
      ip sla schedule operation-number [life {forever | seconds}] [starttime {hh :
     mm[: ss] [month day | day month] | pending | now | after hh : mm : ss}] [ageout
      seconds] [recurring]
     ```

   - **ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period** *schedule-period-range* [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life**{**forever** | *seconds*}] [**starttime**{ *hh:mm*[:*ss*] [*month day* | *day month]* | **pending** | **now** | **after** *hh:mm:ss*}]

     例：

     ```
     switch(config)# ip sla group schedule 1 3,4,6-9
     ```

4. **exit**
5. **show ip sla group schedule**
6. **show ip sla configuration**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	次のいずれかを実行します。 • **ip sla schedule** *operation-number* [**life forever** { | *seconds*}] [**starttime** {*hh : mm*[: *ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh : mm : ss*}] [**ageout** *seconds*] [**recurring**] 例： `ip sla schedule operation-number [life {forever | seconds}] [starttime {hh : mm[: ss] [month day | day month] | pending | now | after hh : mm : ss}] [ageout seconds] [recurring]` • **ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period**	- • 個々の IP SLA 動作の場合のみ： 個々の IP SLA 動作のスケジューリング パラメータを設定します。 • 複数動作スケジューラの場合のみ： スケジューリングされる IP SLA 動作グループ番号と動作番号の範囲をグローバル コンフィギュレーション モードで指定します。

	コマンドまたはアクション	目的
	schedule-period-range [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life**{**forever** | *seconds*}] [**starttime**{ *hh:mm*[*:ss*] [*month day* | *day month]* | **pending** | **now** | **after** *hh:mm:ss*}] 例： `switch(config)# ip sla group schedule 1 3,4,6-9`	
ステップ 4	**exit** 例： `switch(config)# exit`	特権 EXEC モードに戻ります。
ステップ 5	**show ip sla group schedule** 例： `switch# show ip sla group schedule`	（任意）IP SLA グループ スケジュールの詳細を表示します。
ステップ 6	**show ip sla configuration** 例： `switch# show ip sla configuration`	（任意）IP SLA 設定の詳細を表示します。

次のタスク

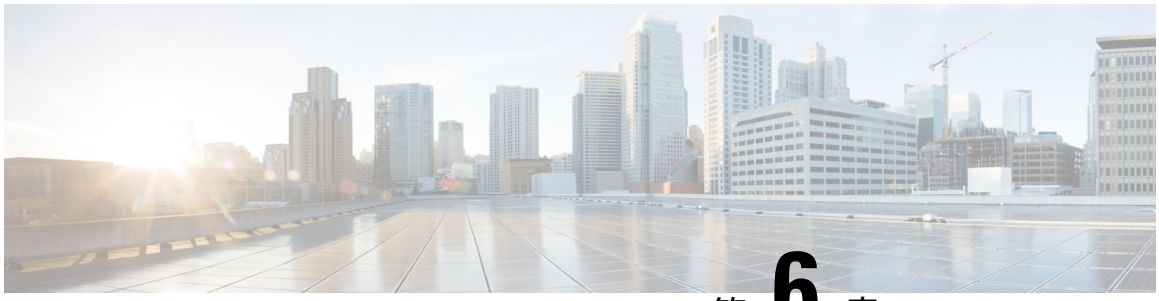
トラップを生成する目的、または別の動作を開始する目的で、動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

IP SLA 動作の結果を表示し、内容を確認するには、**show ip sla statistics** コマンドを使用します。 を実行する前に、ユーザ名がフィギュレーション ファイルに指定されていることを確認してください。サービス レベル契約の基準に対応するフィールドの出力を確認すると、サービス メトリックが許容範囲内であるかどうかを判断する役に立ちます。

UDP エコー動作の構成例

以下に、ただちに開始され、無期限に実行される UDP エコーの IP SLA 動作タイプを構成する例を示します。

```
ip sla 5
udp-echo 172.29.139.134 5000
frequency 30
request-data-size 160
tos 128
timeout 1000
tag FLL-RO
ip sla schedule 5 life forever start-time now
```



第 6 章

IP SLA TCP 接続動作の設定

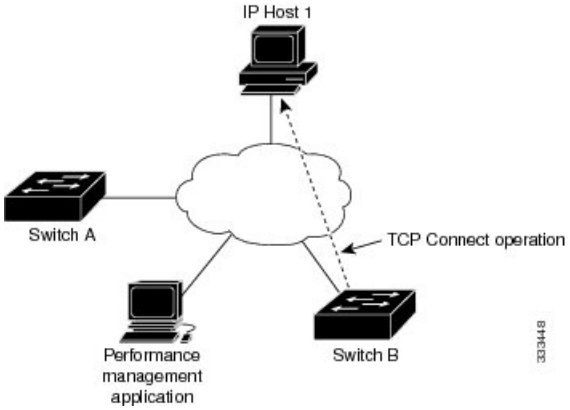
この章では、Cisco スイッチと IPv4 を使用するデバイスの間の、TCP 接続動作の実行に要する応答時間を測定できるように、IP サービス レベル契約（SLA）の TCP 接続動作を構成する方法について説明します。TCP 接続の精度は、宛先のCisco スイッチで IP SLA Responder を使用することによって向上します。この章では、TCP 接続動作の結果を表示して分析し、ネットワーク内のサーバーおよびホストへの接続回数が、IP サービス レベルにどのように影響する可能性があるかを判断する方法についても説明します。TCP 接続動作は、特定のアプリケーションに使用するサーバーの応答時間の測定やサーバーの可用性の接続テストに役立ちます。

この章には、以下の項があります。

TCP 接続動作に関する情報

IP SLA TCP 接続動作は、Cisco スイッチと IP を使用するデバイスの間の TCP 接続動作の実行に要する応答時間を測定します。TCPは、信頼性の高い全二重データ伝送を行うトランスポート層（レイヤ4）インターネットプロトコルです。宛先デバイスは、IPを使用する任意のデバイスまたは IP SLA Responder になります。

次の図では、スイッチ B が送信元 IP SLA デバイスとして設定され、IP ホスト 1 を宛先デバイスとする TCP 接続動作が設定されています。



接続応答時間は、スイッチ B から IP ホスト 1 に TCP 要求メッセージを送信してから、IP ホスト 1 からの応答を受信するまでの時間を測定して算出されます。

TCP 接続の精度は、宛先のシスコ デバイスに IP SLA Responder を使用することによって向上します。宛先スイッチが Cisco スイッチの場合、IP SLA Responder は、指定した任意のポート番号への TCP 接続を確立します。宛先が Cisco IP ホストでない場合は、既知の宛先ポート番号を指定する必要があります（たとえば、FTP には 21、Telnet には 23、HTTP サーバーには 80 を指定）。

シスコ デバイスを使用する場合、TCP 接続動作に IP SLA Responder を使用するかどうかは任意です。シスコ以外のデバイスに IP SLA Responder を設定することはできません。

TCP 接続は、仮想回線の可用性またはアプリケーションの可用性をテストするために使用します。Telnet、SQL、および他のタイプの接続をシミュレーションすることによってサーバーおよびアプリケーションの接続パフォーマンスをテストすると、IP サービス レベルの確認に役立ちます。

IP SLA TCP 接続動作の設定に関する注意事項と制約事項

- キーワードが付いている**show**コマンド**internal**はサポートされていません。

IP SLA パケットの CoPP の構成

IP SLA 動作を大規模なスケールで使用する場合、IP SLA パケットのパススルーを許可する特定の CoPP 構成が必要になる場合があります。IP SLA ではユーザー定義の UDP ポートを使用するため、コントロール プレーンへのすべての IP SLA パケットを許可する手段がありません。ただし、IP SLA が使用できる接続先/送信元ポートのそれぞれを指定することはできます。

IP SLA プローブ数の検証済みの拡張性に関する詳細については、*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*を参照してください。

以下に、IP SLA パケットのパススルーを許可する CoPP 構成例を示します。この例では、接続先ポートと送信元ポートが 6500 ～ 7000 の範囲であることを前提としています。この例では、

「insert-before」が指定されていない場合、「class-default」の後に「copp-ipsla」が追加されます。

（注） 次の構成例は、プラットフォーム/ハードウェアタイプによって異なる場合があります。IP ACL および CoPP の設定の詳細については、『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』を参照してください。

```
ip access-list acl-sla-allow
  10 remark ### ALLOW SLA control packets from 1.1.1.0/24
  20 permit udp 1.1.1.0/24 any eq 1967
  30 remark ### ALLOW SLA data packets from 1.1.1.0/24 using ports 6500-7000
  40 permit udp 1.1.1.0/24 any range 6500 7000

class-map type control-plane match-any copp-ipsla
  match access-group name acl-sla-allow

policy-map type control-plane Custom-copp-policy-strict
    class copp-ipsla insert-before Custom-copp-class-l2-default
    police cir 1500 kbps

control-plane
  service-policy input Custom-copp-policy-strict

switch# show policy-map interface control-plane | be copp-ipsla
    class-map copp-ipsla (match-any)
      match access-group name acl-sla-allow
      set cos 7
      police cir 1500 kbps , bc 32000 bytes
      module 1 :
        transmitted 0 bytes;
        dropped 0 bytes;

    class-map Custom-copp-class-l2-default (match-any)
      match access-group name Custom-copp-acl-mac-undesirable
      set cos 0
      police cir 400 kbps , bc 32000 bytes
      module 1 :
        transmitted 0 bytes;
        dropped 0 bytes;

    class-map class-default (match-any)
      set cos 0
      police cir 400 kbps , bc 32000 bytes
      module 1 :
        transmitted 122 bytes;
        dropped 0 bytes;
```

Netstack ポート範囲の一致

IP SLA は、ローカルのネットスタック ポート範囲内のポートのみを受け入れます。プローブの設定で使用される送信元ポートと接続先ポートは、SLA 送信側と SLA レスポンダでサポートされている netstack ポートと一致している必要があります。

以前のバージョンからバージョン 9.3(1) 以降のバージョンに ISSU を実行する場合は、SSH ポートなどのユーザー定義ポートの機能が次の表に記載されている範囲内にあることを確認してください。

表 *11 : ISSU* のポート範囲

バージョン	デフォルトのポート範囲
9.3(1)	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 63535） nat ポート範囲（63536 ～ 65535）
9.3(2)	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 63535） nat ポート範囲（63536 ～ 65535）
9.3(3) 以降	Kstack ローカル ポート範囲（15001 ～ 58000） Netstack ローカル ポート範囲（58001 ～ 60535） nat ポート範囲（60536 - 65535）

show sockets local-port-range コマンドを使用すれば コマンドは、送信側/応答側のポート範囲を表示します。

以下は、netstack ポート範囲を表示する例です。

```
switch# show sockets local-port-range

Kstack local port range (15001 - 22002)
Netstack local port range (22003 - 65535)
```

宛先デバイスでの IP SLA Responder の設定

この項では、接続先デバイスで IP SLA Responder を設定する方法について説明します。

始める前に

IP SLA Responder を使用する場合は、応答側として使用するネットワーキング デバイスがシスコ デバイスであり、そのデバイスにネットワークを介して接続できることを確認します。

手順の概要

1. **enable**

2. **configure terminal**
3. **feature sla responder**
4. 次のいずれかを実行します。
   - **ip sla responder**

     例：

     ```
     switch(config)# ip sla responder
     ```

   - **ip sla responder tcp-connect ipaddress** *ip-address* **port** *port*

     例：

     ```
     switch(config)# ip sla responder tcp-connect ipaddress 172.29.139.132 port 5000
     ```

5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla responder** 例： `switch(config)# feature sla responder`	IP SLA のレスポンダ機能を有効にします。

ステップ 4	次のいずれかを実行します。
	• **ip sla responder**
	例：
	`switch(config)# ip sla responder`
	• **ip sla responder tcp-connect ipaddress** *ip-address* **port** *port*
	例：
	`switch(config)# ip sla responder tcp-connect ipaddress 172.29.139.132 port 5000`
		-
		• （任意）送信元からの制御メッセージに応じて、Cisco デバイスにおける IP SLA Responder 機能を一時的に有効にします。
		• （任意）送信元でプロトコル制御がディセーブルである場合にのみ必須です。このコマンドは、指定の IP アドレスおよびポートで IP SLA Responder 機能を永続的に有効にします。
		制御は、デフォルトでイネーブルになります。

ステップ 5	**exit** 例： `switch(config)# exit`	（任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

送信元デバイスでのTCP接続動作の設定およびスケジューリング

ここでは、送信元デバイスでの TCP 接続動作を構成し、スケジュールする方法について説明します。

送信元デバイスの TCP 接続動作を構成し、スケジュールするには、次のいずれか 1 つのタスクだけを実行します。

- 送信元デバイスでの基本 TCP 接続動作の構成とスケジューリング
- 送信元デバイスでのオプション パラメータを使用した TCP 接続動作の構成とスケジューリング

送信元デバイスでの基本のTCP接続動作の設定およびスケジューリング

ここでは、送信元デバイスでの基本 TCP 接続動作を設定およびスケジュールする方法について説明します。

（注） IP SLA レスポンダが宛先 IP アドレスとポートで永続的に有効になっている場合は、 **control** を使用します。 **disable tcp-connect** を使ったキーワード制御メッセージを無効にするコマンド。

ヒント
- IP SLA 動作が実行せず、統計情報が生成されていない場合は、動作の設定に **verify-data** コマンドを追加して（IP SLA 構成モードで設定）、データ検証を有効にします。イネーブルになると、各動作の応答が破損していないかどうかがチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- **debug ip sla sender trace** コマンドを使用し、 および **debug ip sla sender error** IP SLA 動作に関する問題をトラブルシューティングするコマンドです。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla sender**
4. **ip sla** *operation-number*
5. **tcp-connect** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **source-port***port-number*] [**control** {**enable** | **disable**}]

6. **frequency** *seconds*
7. **exit**
8. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh*:*mm*:*ss*] [**ageout** *seconds*] [**recurring**]
9. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 4	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 5	**tcp-connect** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **source-port***port-number*] [**control** {**enable** | **disable**}] 例： `switch(config-ip-sla)# tcp-connect 172.29.139.132 5000`	TCP 接続動作を定義し、IP SLA TCP コンフィギュレーション モードを開始します。 送信元スイッチとターゲット スイッチの両方で IP SLA 制御プロトコルを無効にする場合のみ、**control disable** キーワードの組み合わせを使用します。
ステップ 6	**frequency** *seconds* 例： `switch(config-ip-sla-tcp)# frequency 60`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 7	**exit** 例： `switch(config-ip-sla-tcp)# exit`	IP SLA TCP 構成モードを終了し、グローバル構成モードに戻ります。

	コマンドまたはアクション	目的
ステップ 8	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh*:*mm*:*ss*] [**ageout** *seconds*] [**recurring**] 例： `switch(config)# ip sla schedule 10 start-time now life forever`	個々の IP SLA 動作のスケジューリング パラメータを設定します。
ステップ 9	**exit** 例： `switch(config)# exit`	（任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、即時に開始されて無期限に実行される TCP 接続の IP SLA 動作タイプを構成する例を示します。

```
feature sla sender
ip sla 9
 tcp-connect 172.29.139.132 5000
 frequency 10
!
ip sla schedule 9 life forever start-time now
```

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

IP SLA 動作の結果を表示し、内容を確認するには、**show ip sla statistics** コマンドを使用します。 を実行する前に、ユーザ名がフィギュレーション ファイルに指定されていることを確認してください。サービス レベル契約の基準に対応するフィールドの出力を確認すると、サービス メトリックが許容範囲内であるかどうかを判断する役に立ちます。

送信元デバイスでのオプションパラメータを使用したTCP接続動作の構成とスケジューリング

ここでは、オプション パラメータを使用して、送信元デバイスでの TCP 接続動作を設定し、スケジュールする方法について説明します。

（注） IP SLA Responder が宛先 IP アドレスとポートで永続的に有効になっている場合は、 **control disable** キーワードを **tcp-connect** コマンドで使用して、制御メッセージを無効にします。

ヒント
- IP SLA 動作が実行されておらず、統計を生成していない場合は、 **verify-data** コマンドを動作の構成に追加して（IP SLA 構成モードで設定）、データ検証を有効にします。イネーブルになると、各動作の応答が破損していないかどうかがチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- **debug ip sla trace** コマンドを使用し、 および **debug ip sla error** コマンドは、IP SLA 動作に関する問題のトラブルシューティングを行うために使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla sender**
4. **ip sla** *operation-number*
5. **tcp-connect** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **source-port** *port-number*] [**control** {**enable** | **disable**}]
6. **history buckets-kept** *size*
7. **history distributions-of-statistics-kept** *size*
8. **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*]
9. **history filter** {**none** | **all** | **overThreshold** | **failures**}
10. **frequency** *seconds*
11. **history hours-of-statistics-kept** *hours*
12. **history lives-kept** *lives*
13. **owner** *owner-id*
14. **history statistics-distribution-interval** *milliseconds*
15. **tag** *text*
16. **threshold** *milliseconds*
17. **timeout** *milliseconds*
18. **tos** *number*
19. **exit**
20. **ip sla schedule** *operation-number* [**life** {**forever**| *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh*:*mm*:*ss*] [**ageout** *seconds*] [**recurring**]
21. **exit**
22. **show ip sla configuration** [*operation-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 4	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 5	**tcp-connect** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*} **source-port** *port-number*] [**control** {**enable** | **disable**}] 例： `switch(config-ip-sla)# tcp-connect 172.29.139.132 5000`	TCP 接続動作を定義し、IP SLA TCP コンフィギュレーション モードを開始します。 送信元スイッチとターゲット スイッチの両方で IP SLA 制御プロトコルを無効にする場合のみ、**control disable** キーワードの組み合わせを使用します。
ステップ 6	**history buckets-kept** *size* 例： `switch(config-ip-sla-tcp)# history buckets-kept 25`	（任意）IP SLA 動作のライフタイム中に保持する履歴バケット数を設定します。
ステップ 7	**history distributions-of-statistics-kept** *size* 例： `switch(config-ip-sla-tcp)# history distributions-of-statistics-kept 5`	（任意）IP SLA 動作中にホップ単位で保持する統計情報の配信数を設定します。
ステップ 8	**history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*] 例：	（任意）IP SLA 動作に対する拡張履歴収集をイネーブルにします。

	コマンドまたはアクション	目的
	`switch(config-ip-sla-tcp)# history enhanced interval 900 buckets 100`	
ステップ 9	**history filter** {**none** | **all** | **overThreshold** | **failures**} 例： `switch(config-ip-sla-tcp)# history filter failures`	（任意）IP SLA 動作の履歴テーブルに格納する情報のタイプを定義します。
ステップ 10	**frequency** *seconds* 例： `switch(config-ip-sla-tcp)# frequency 60`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 11	**history hours-of-statistics-kept** *hours* 例： `switch(config-ip-sla-tcp)# history hours-of-statistics-kept 4`	（任意）IP SLA 動作の統計情報を保持する時間数を設定します。
ステップ 12	**history lives-kept** *lives* 例： `switch(config-ip-sla-tcp)# history lives-kept 5`	（任意）IP SLA 動作の履歴テーブルに格納するライフ数を設定します。
ステップ 13	**owner** *owner-id* 例： `switch(config-ip-sla-tcp)# owner admin`	（任意）IP SLA 動作の簡易ネットワーク管理プロトコル（SNMP）所有者を設定します。
ステップ 14	**history statistics-distribution-interval** *milliseconds* 例： `switch(config-ip-sla-tcp)# history statistics-distribution-interval 10`	（任意）IP SLA 動作で維持する各統計情報の配信間隔を設定します。
ステップ 15	**tag** *text* 例： `switch(config-ip-sla-tcp)# tag TelnetPollServer1`	（任意）IP SLA 動作のユーザー指定 ID を作成します。
ステップ 16	**threshold** *milliseconds* 例： `switch(config-ip-sla-tcp)# threshold 10000`	（任意）IP SLA 動作によって作成されるネットワーク モニタリング統計情報を計算するための上限しきい値を設定します。

	コマンドまたはアクション	目的
ステップ 17	**timeout** *milliseconds* 例： `switch(config-ip-sla-tcp)# timeout 10000`	（任意）IP SLA 動作がその要求パケットからの応答を待機する時間を設定します。
ステップ 18	**tos** *number* 例： `switch(config-ip-sla-jitter)# tos 160` 例：	（任意）IPv4 ネットワークに限り、IP SLA 動作の IPv4 ヘッダーの ToS バイトを定義します。
ステップ 19	**exit** 例： `switch(config-ip-sla-tcp)# exit`	TCP コンフィギュレーション サブモードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 20	**ip sla schedule** *operation-number* [**life** {**forever**| *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh*:*mm*:*ss*] [**ageout** *seconds*] [**recurring**] 例： `switch(config)# ip sla schedule 10 start-time now life forever`	個々の IP SLA 動作のスケジューリング パラメータを設定します。
ステップ 21	**exit** 例： `switch(config)# exit`	（任意）グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 22	**show ip sla configuration** [*operation-number*] 例： `switch# show ip sla configuration 10`	（任意）すべての IP SLA 動作または指定した IP SLA 動作に関する設定値を、すべてのデフォルト値を含めて表示します。

例

次に、TCP 接続動作番号 10 の IP SLA パラメータをすべて（デフォルトを含む）設定する例を示します。

```
switch# show ip sla configuration 10
IP SLAs Infrastructure Engine-III
Entry number: 10
Owner: admin
Tag: TelnetPollServer1
Operation timeout (milliseconds): 10000
Type of operation to perform: tcp-connect
```

```
Target address/Source address: 101.101.101.1/0.0.0.0
Target port/Source port: 5000/0
Type Of Service parameter: 0xa0
Vrf Name: default
Control Packets: enabled
Schedule:
   Operation frequency (seconds): 60  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : FALSE
   Randomly Scheduled : FALSE
   Life (seconds): Forever
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 10000
Distribution Statistics:
   Number of statistic hours kept: 4
   Number of statistic distribution buckets kept: 5
   Statistic distribution interval (milliseconds): 10
Enhanced History:
   Aggregation Interval:900 Buckets: 100
History Statistics:
   Number of history Lives kept: 0
   Number of history Buckets kept: 25
   History Filter Type: Failures
```

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

IP SLA 動作の結果を表示し、内容を確認するには、**show ip sla statistics** コマンドを使用します。 を実行する前に、ユーザ名がフィギュレーション ファイルに指定されていることを確認してください。サービス レベル契約の基準に対応するフィールドの出力を確認すると、サービス メトリックが許容範囲内であるかどうかを判断する役に立ちます。

TCP 接続動作の構成例

次に、「IP SLA TCP 接続動作に関する情報」の項の図「TCP 接続動作」に示されているように、スイッチ B から IP ホスト 1（IP アドレス 10.0.0.1）の Telnet ポート（TCP ポート 23）への TCP 接続動作を設定する例を示します。動作は、ただちに開始されるようにスケジューリングされます。この例では、送信元（スイッチB）で制御プロトコルが無効になっています。IP SLA は制御プロトコルを使用して、ターゲット ポートを一時的に有効にするように IP SLA レスポンダに通知します。このアクションにより、レスポンダは TCP 接続動作に応答できます。この例では、ターゲットがスイッチではなく、既知のTCPポートが使用されているため、制御メッセージを送信する必要はありません。

スイッチ A の設定

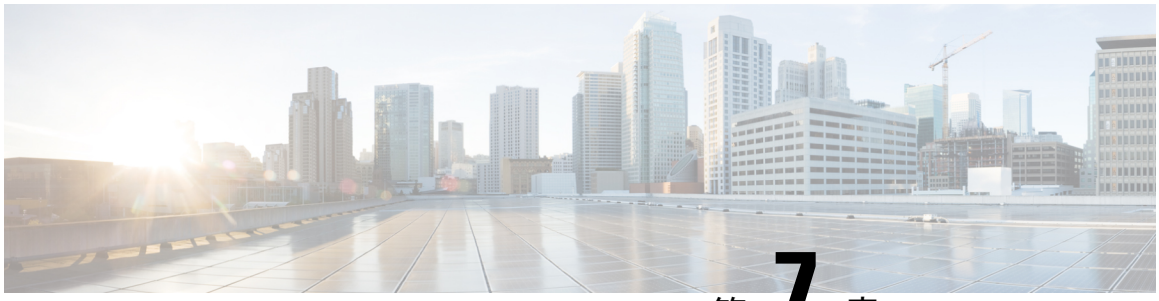
```
configure terminal
 feature sla responder
 ip sla responder tcp-connect ipaddress 10.0.0.1 port 23
```

スイッチ B の設定

```
configure terminal
 feature sla sender
 ip sla 9
  tcp-connect 10.0.0.1 23 control disable
  frequency 30
  tos 128
  timeout 1000
  tag FLL-RO
 ip sla schedule 9 start-time now
```

次に、特定のポート（ポート 21）を使用し、IP SLA レスポンダを使用せずに TCP 接続動作を構成する例を示します。動作は、ただちに開始され、無期限に実行するようスケジューリングされます。

```
configure terminal
 feature sla sender
 ip sla 9
  tcp-connect 173.29.139.132 21 control disable
  frequency 30
 ip sla schedule 9 life forever start-time now
```



第 7 章

IP SLA HTTP 動作の構成

この章では、HTTP IP サービス レベル アグリーメント（IP SLA）動作を構成する方法について説明します。

この章は、次の項で構成されています。

IP SLA HTTP 動作の構成

この章では、Cisco デバイスと HTTP サーバーの間で Web ページを取得するための応答時間をモニタするように、IP サービス レベル契約（SLA）HTTP 動作を設定する方法について説明します。IP SLA FTP 動作は通常の GET要求だけをサポートします。

IP SLA HTTP 動作について

HTTP 要求はプロキシ サーバーを経由して行うことができます。

HTTP 動作は、シスコ デバイスと HTTP サーバーの間で Web ページを取得するためのラウンドトリップ時間（RTT）を測定します。HTTP サーバー応答時間の測定は次の 3 つの RTT から構成されます。

- DNS ルックアップ：ドメイン名ルックアップの実行に要する RTT。
- TCP 接続：HTTP サーバーへの TCP 接続の実行に要する RTT。
- HTTP トランザクション時間：要求を送信し、HTTP サーバーからの応答の取得に要する RTT。この動作はホーム HTML ページだけを取得します。

HTTP 操作は、最初に DNS 操作を実行し、DNS RTT を測定します。ドメイン名が見つかったら、HTTP 動作は、適切な HTTP サーバーに対する TCP 接続動作を実行します。次に、HTTP

操作は TCP 接続 RTT を測定します。最後に、HTTP 操作は HTTP 要求を送信し、HTTP サーバーからホーム HTML ページを取得します。次に、HTTP 操作は RTT を測定して、ホーム HTML ページを取得します。HTTP 操作は最後に、「最初のバイトまでの時間」と呼ばれる別の測定を行います。この測定によって、TCP 接続動作の開始から HTTP 操作により取得された最初の HTML バイトを検出するまでの時間が測定されます。総 HTTP RTT は、DNS RTT、TCP 接続 RTT、および HTTP RTT の合計です。合計の HTTP RTT を調べることにより、Web ページの取得にかかった RTT を判断して、Web サーバーのパフォーマンス レベルをモニタするのに役立ちます。

GET 要求の場合、IP SLA は指定された URL に基づいて要求の形式を設定します。

IP SLA HTTP 動作の制約事項

IP SLA HTTP 動作には、次の制限があります。

- IP SLA HTTP オペレーションは、Cisco NX-OS リリース 7.0(3)I6(1) 以降の Cisco Nexus 9300 および 9500 シリーズ スイッチでは、HTTP GET プローブのみをサポートします。
- 頻度を 60 秒未満に設定すると、送信されるパケット数が増加します。しかしこのことは、スケジュールされた動作の開始時刻が同じ場合、IP SLA 動作のパフォーマンスに悪影響を与える可能性があります。

基本的な HTTP GET 動作の構成

HTTP GET メソッドは、Request-URL で識別される情報を (エンティティの形式で) 取得します。

手順の概要

1. **configure terminal**
2. **ip sla operation-number**
3. **http**{**get** | **url** [**version** *version-number*] [**source-ip**{*ip-address* | *hostname*}] [**source-port** *port-number*] [**cache**{**enable** | **disable**}] [**proxy** *proxy-url*]
4. **frequency** *seconds*
5. **end**

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 terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	**ip sla operation-number** 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 3	**http**{**get** | **url** [**version** *version-number*] [**source-ip**{*ip-address* | *hostname*}] [**source-port** *port-number*] [**cache**{**enable** | **disable**}] [**proxy** *proxy-url*] 例： `switch(config-ip-sla-http)# http get http://198.133.219.25`	HTTP 動作を定義し、IP SLA コンフィギュレーション モードを開始します。
ステップ 4	**frequency** *seconds* 例： `switch(config-ip-sla-http)# frequency 90`	（任意）指定した IP SLA HTTP 動作を繰り返す間隔を設定します。IP SLA HTTP 動作のデフォルトの最小頻度値は 60 秒です。
ステップ 5	**end** 例： `switch(config-ip-sla-http)# end`	IP SLA 構成モードを終了します。

オプション パラメータを使用した HTTP GET 動作の構成

手順の概要

1. **configure terminal**
2. **ip sla operation-number**
3. **http**{**get** | **url** [**version** *version-number*] [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**cache**{**enable** | **disable**}] [**proxy** *proxy-url*]
4. **history buckets-kept** *size*
5. **history distributions-of-statistics-kept** *size*
6. **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*]
7. **history filter** { **none**| **all** | **overThreshold** | **failures**}
8. **frequency** *seconds*
9. **history hours-of-statistics-kept** *hours*
10. **history live-kept** *lives*
11. **owner owner-id**
12. **history statistics-distribution-interval** *milliseconds*
13. **tag text**
14. **threshold** *milliseconds*
15. **timeout** *milliseconds*
16. **tos** *number*
17. **end**

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 terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**ip sla operation-number** 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 3	**http**{**get** | **url** [**version** *version-number*] [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**cache**{**enable** | **disable**}] [**proxy** *proxy-url*] 例： `switch(config-ip-sla)# http get http://198.133.219.25`	HTTP 動作を定義し、IP SLA コンフィギュレーション モードを開始します。
ステップ 4	**history buckets-kept** *size* 例： `switch(config-ip-sla-http)# history buckets-kept 25`	（任意）IP SLA 動作のライフタイム中に保持する履歴バケット数を設定します。
ステップ 5	**history distributions-of-statistics-kept** *size* 例： `switch(config-ip-sla-http)# history distribution-of-statistics-kept 5`	（任意）IP SLA 動作中にホップ単位で保持する統計情報の配信数を設定します。
ステップ 6	**history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*] 例： `switch(config-ip-sla-http)# history enhanced interval 900 buckets 100`	（任意）IP SLA 動作に対する拡張履歴収集をイネーブルにします。
ステップ 7	**history filter** { **none**| **all** | **overThreshold** | **failures**} 例： `switch(config-ip-sla-http)# history filter failures`	（任意）IP SLA 動作の履歴テーブルに格納する情報のタイプを定義します。
ステップ 8	**frequency** *seconds* 例： `switch(config-ip-sla-http)# frequency 90`	（任意）指定した IP SLA HTTP 動作を繰り返す間隔を設定します。IP SLA HTTP 動作のデフォルトの最小頻度値は 60 秒です。
ステップ 9	**history hours-of-statistics-kept** *hours* 例：	（任意）IP SLA 動作を継続する時間の長さを設定します。

	コマンドまたはアクション	目的
	`switch(config-ip-sla-http)# history hours-of-statistics-kept 4`	
ステップ 10	**history live-kept** *lives* 例： `switch(config-ip-sla-http)# history lives-kept 5`	（任意）IP SLA 動作を保持するライフ数を設定します。
ステップ 11	**owner owner-id** 例： `switch(config-ip-sla-http)# owner admin`	（任意）IP SLA 動作の簡易ネットワーク管理プロトコル（SNMP）を構成します。
ステップ 12	**history statistics-distribution-interval** *milliseconds* 例： `switch(config-ip-sla-http)# history statistics-distribution-interval 10`	（任意）IP SLA 動作で維持する各統計情報の配信間隔を設定します。
ステップ 13	**tag text** 例： `switch(config-ip-sla-http)# tag TelnetPollServer1`	（任意）IP SLA 動作のユーザー指定 ID を作成します。
ステップ 14	**threshold** *milliseconds* 例： `switch(config-ip-sla-http)# threshold 10000`	（任意）IP SLA 動作によるネットワーク モニタリング統計情報を計算するための上限しきい値を設定します。
ステップ 15	**timeout** *milliseconds* 例： `switch(config-ip-sla-http)# timeout 10000`	（任意）IP SLA 動作の要求パケットからの最大応答時間を設定します。
ステップ 16	**tos** *number* 例： `switch(config-ip-sla-http)# tos 160`	（任意）IP SLA 動作の IP ヘッダー内のタイプ オブ サービス（ToS）バイトを定義します。
ステップ 17	**end** 例： `switch(config-ip-sla-http)# end`	IP SLA 構成モードを終了します。

IP SLA 動作のスケジューリング

始める前に

- スケジュールする前に、すべての IP サービス レベル アグリーメント（SLA）操作を構成します。

- 複数動作グループでスケジュールされたすべての動作は、頻度が同じでなければなりません。
- 複数動作グループに追加する動作 ID 番号のリストは、カンマ（,）を含めて最大 125 文字に制限されます。
- 動作をスケジュールする前に、次のことを確認してください。
  - スケジューリングする前に、IP SLA 動作を設定しておきます。
  - 複数動作グループでスケジュールされたすべての動作は、頻度が同じでなければなりません。
  - 複数動作グループに追加する動作 ID 番号のリストは、カンマ（,）を含めて最大 125 文字に制限する必要があります。

手順の概要

1. **configure terminal**
2. スケジュールする IP SLA 動作の数に基づいて、次のいずれかのコマンドを選択します。
   - **ip sla** schedule **operation number** [**life**{**forever** | **seconds**}] [**start-time**{[*hh:mm:ss*] [**month day** | **day month**| **pending** | **now** | **after** [*hh:mm:ss*}| [**ageout seconds**] [**recurring**]
   - **ip sla group schedule** *group-operation-number operation-id-numbers* {**schedule-period** *schedule-period-range* | **schedule-together**} [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life** {**foreever**}] **start-time** {*hh:mm* [*:ss*] [**month day** | **day month**] |**pending**|**now** |**after** *hh:mm* [*:ss*]}]
3. **show ip sla group schedule**
4. **show ip sla group configuration**
5. **end**

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 terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	スケジュールする IP SLA 動作の数に基づいて、次のいずれかのコマンドを選択します。 • **ip sla** schedule **operation number** [**life**{**forever** | **seconds**}] [**start-time**{[*hh:mm:ss*] [**month day** | **day month**| **pending** | **now** | **after** [*hh:mm:ss*}| [**ageout seconds**] [**recurring**] • **ip sla group schedule** *group-operation-number operation-id-numbers* {**schedule-period** *schedule-period-range* | **schedule-together**} [**ageout**	最初のコマンドでは、個々の IP SLA 動作のスケジューリング パラメータを設定しています。 二番目のコマンドでは、複数動作スケジューラ用に IP SLA 動作グループ番号と動作番号の範囲を指定しています。

	コマンドまたはアクション	目的

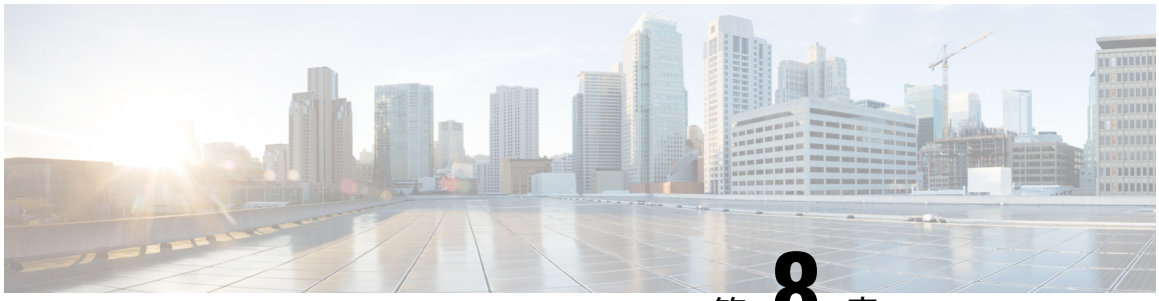
	seconds] [**frequency** *group-operation-frequency*] [**life** {**foreever**}] **start-time** {*hh:mm* [*:ss*] [**month day** | **day month**] |**pending**|**now** |**after** *hh:mm* [*:ss*]}]
	例：
	`switch (config-ip-sla-http)# ip sla schedule 10 life forever start-time now`
	`switch (config-ip-sla-http)# ip sla group schedule 10 life schedule-period frequency`
	`switch (config-ip-sla-http)# ip sla group schedule 1.3.4.6-9 life forever start-time now`
	`switch (config-ip-sla-http)# ip sla group schedule 1.3.4.6-9 schedule-period 50 frequency range 80-100`	

ステップ 3	**show ip sla group schedule** 例： `switch(config-ip-sla-http)# show ip sla group schedule`	（任意）IP SLA グループ スケジュールの詳細を表示します。
ステップ 4	**show ip sla group configuration** 例： `switch(config-ip-sla-http)# show ip sla group configuration`	（任意）IP SLA 設定の詳細を表示します。
ステップ 5	**end** 例： `switch(config-ip-sla-http)# end`	IP SLA 構成モードを終了します。

トラブルシューティングのヒント

IP SLA 動作で統計が生成されない場合は、verify-data コマンドを使用して構成します。これにより、操作ごとに応答の破損がないかチェックできます。IP SLA 動作が実行されていないことを確認してください。そうでないと、verify-data コマンドによって不要なオーバーヘッドが生成されます。

IP SLA 動作に関する問題のトラブルシューティングを行うには、debug ip sla trace コマンドと debug ip sla error コマンドを使用します。



第 8 章

複数動作スケジューラの構成

この章では、IP サービス レベル契約（IP SLA）の複数動作スケジューラを使用して複数の動作をスケジューリングする方法について説明します。

この章は、次の項で構成されています。

IP SLA 複数動作スケジューラに関する情報

IP SLA 動作の通常のスケジューリングでは、一度に 1 つの動作をスケジューリングできます。大規模なネットワークで、何千もの IP SLA 動作によりネットワーク パフォーマンスをモニタする場合、通常のスケジューリング（各動作を個別にスケジューリングする方法）は、非効率的であり、時間がかかります。

複数動作のスケジューリングでは、コマンドライン インターフェイス（CLI）または CISCO RTTMON-MIB による単一のコマンドを使用して、複数の IP SLA 動作をスケジューリングすることができます。この機能では、これらの動作を均等な時間間隔で実行するようにスケジューリングすることで、IP SLA モニタリング トラフィックの量を制御できます。スケジューリン

グされる動作 ID 番号、およびすべての IP SLA 動作が開始されなければならない時間の範囲を指定する必要があります。この機能は、指定したタイム フレームにおいて等間隔で自動的に IP SLA 動作を分散します。動作の間隔（開始間隔）が計算されて、動作が開始されます。このように IP SLA 動作を分散することで、CPU の使用を最小限に抑えることが可能になり、ネットワークのスケーラビリティが向上します。

IP SLA 複数動作スケジューリング機能では、次の設定パラメータを使用して、複数の IP SLA 動作を 1 つのグループとしてスケジュールできます。

- グループ動作番号（Group operation number）：スケジューリングされる IP SLA 動作のグループ設定またはグループ スケジュール番号。
- 動作 ID 番号（Operation ID numbers）：スケジューリングされる動作グループの IP SLA 動作 ID 番号のリスト。
- スケジュール期間：IP SLA 動作グループがスケジューリングされる時間。
- エージアウト：情報をアクティブに収集していないときに、メモリ内に動作を維持する時間。デフォルトでは、動作はメモリに永久に保持されます。
- 頻度（Frequency）：各 IP SLA 動作が再開されるまでの時間。頻度オプションを指定すると、グループに属しているすべての動作の動作頻度が上書きされます。頻度オプションが指定されていない場合、各動作の頻度は、スケジュール期間の値に設定されます。
- ライフ（Life）：動作が情報をアクティブに収集する時間。無期限に実行されるように動作を設定できます。デフォルトでは、動作のライフタイムは 1 時間です。
- 開始時間：動作が情報の収集を開始する時間。すぐに動作を開始するように指定するか、時間、分、秒、日、月を使用して、絶対的な開始時刻に動作を開始するように指定できます。

IP SLA 複数動作スケジューリング機能では、終了せずに実行できる最大動作数をスケジューリングします。ただし、この機能は、すでに実行されている IP SLA 動作や、設定されていないため存在しない動作はスキップします。動作の総数は、不明またはすでに実行されている動作の数に関係なく、コマンドで指定された動作の数に基づいて計算されます。IP SLA 複数動作スケジューリング機能では、アクティブな動作および不明な動作の数を示すメッセージが表示されます。ただし、これらのメッセージが表示されるのは、設定されていないまたはすでに実行されている動作をスケジューリングした場合だけです。

複数の IP SLA 動作をスケジュールする場合の主な利点は、スケジュールされた期間にわたって動作を均一に分散することで、ネットワークの負荷が低減されることです。この分散はより一貫したモニタリングのカバレッジを実現するのに役立ちます。60 秒のスケジュール期間中の同じ 1 秒の間隔以内で 60 個の動作が開始される場合を考えてみます。60 個すべての動作が開始した後にネットワークの障害が 30 秒間発生した場合、それらの動作が再び開始される時間（この障害の 30 秒後）になる前にネットワークが復旧すると、この障害は 60 個のいずれの動作でも検出されません。一方、60 個の動作が 60 秒のスケジュール期間にわたって 1 秒間隔で均等に分散された場合は、一部の動作でこのネットワーク障害が検出されます。逆に、60 個すべての動作がアクティブな時点でネットワーク障害が発生すると、60 個のすべての動作が失敗するため、障害は実際よりも重大であると示される可能性があります。

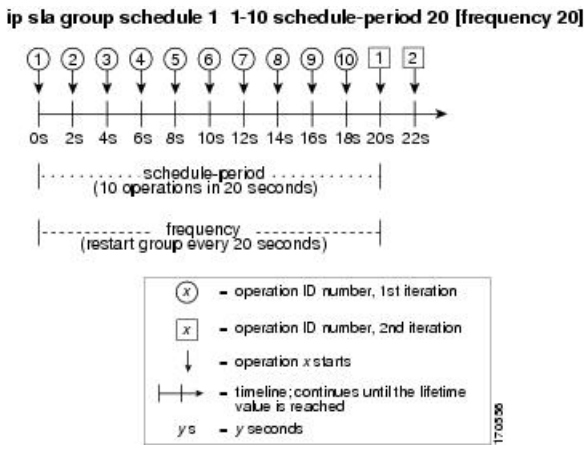
同じタイプの動作では、IP SLA 複数動作スケジューリングに同じ頻度を使用してください。頻度を指定しない場合、デフォルトの頻度はスケジュール期間と同じになります。スケジュール期間は、指定されたすべての動作が実行される必要がある期間です。

IP SLA 複数動作スケジューリングのデフォルトの動作

IP SLA 複数動作スケジューリング機能では、複数の IP SLA 動作を 1 つのグループとしてスケジューリングできます。

次の図に、動作 1 から動作 10 を含む動作グループ 1 のスケジューリングを示します。動作グループ 1 のスケジュール期間は 20 秒です。したがって、このグループ内のすべての動作が 20 秒の期間内に等間隔で開始されます。デフォルトでは、頻度は、設定されたスケジュール期間と同じ値に設定されます。次の図に示すように、頻度はデフォルトで 20 に設定されます。頻度を設定するかどうかは任意です。

図 *3:* スケジュール期間が頻度と等しい：デフォルトの動作



この例では、動作グループ 1 内の最初の動作（動作 1）が 0 秒に開始します。動作グループ 1 内の 10 個すべての動作（動作 1 ～ 10）が、20 秒のスケジュール期間内に開始される必要があります。各 IP SLA 動作の開始時間は、スケジュール期間を動作の数で割ることにより（20 秒が 10 個の動作で割られる）、スケジュール期間にわたって均等に分散されます。したがって、各動作は前の動作の 2 秒後に開始されます。

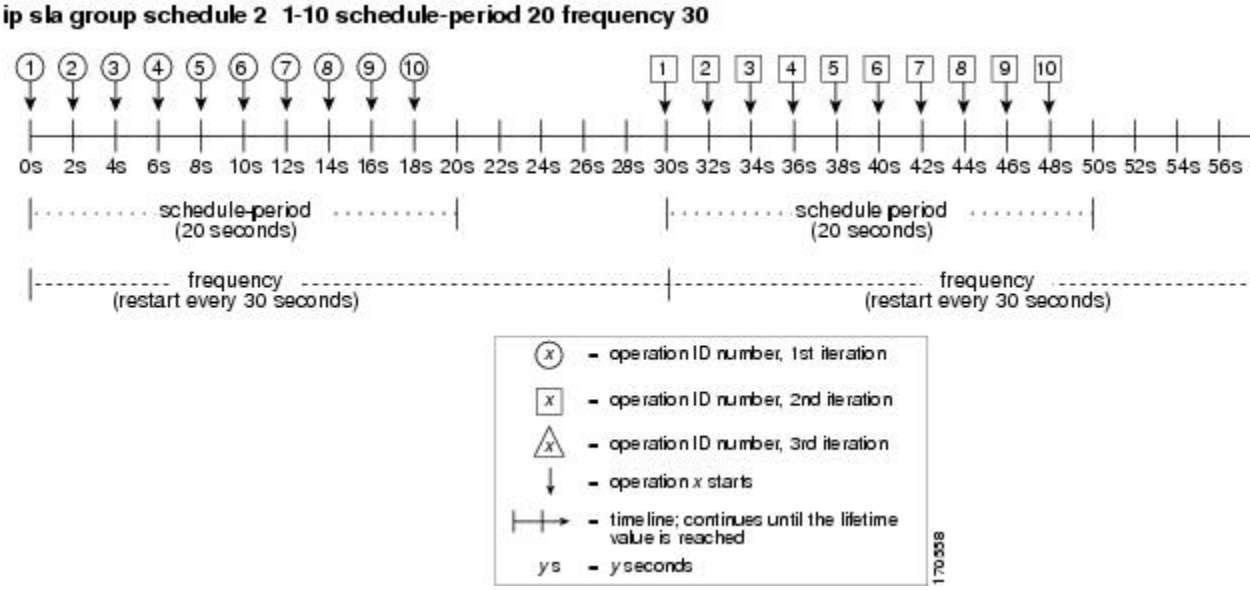
頻度は、動作グループが再開されるまで（繰り返されるまで）の経過時間です。頻度が指定されていない場合、その頻度は、スケジュール期間の値に設定されます。図に示した例では、動作グループ 1 が 20 秒ごとに繰り返し開始されます。この設定では、指定されたスケジュール期間にわたって動作の最適な分割（間隔）が得られています。

スケジュール期間が頻度よりも小さい場合の IP SLA 複数動作スケジューリング

頻度の値は、スケジュール グループが再開されるまでに経過する時間です。スケジュール期間が頻度よりも小さい場合、動作が開始されない期間が出てきます。

次の図に、動作グループ 2 内の動作 1 から動作 10 のスケジューリングを示します。動作グループ 2 のスケジュール期間は 20 秒、頻度は 30 秒です。

図 *4:* スケジュール期間が頻度よりも小さい場合



この例では、動作グループ 2 内の最初の動作（動作 1）が 0 秒に開始します。動作グループ 2 内の 10 個すべての動作（動作 1 ～ 10）が、20 秒のスケジュール期間内に開始される必要があります。各 IP SLA 動作の開始時間は、スケジュール期間を動作の数で割ることにより（20 秒が 10 個の動作で割られる）、スケジュール期間にわたって均等に分散されます。したがって、各動作は前の動作の 2 秒後に開始されます。

動作グループ 2 の最初の繰り返しでは、動作 1 が 0 秒で開始され、最後の動作（動作 10）が 18 秒で開始されます。ただし、グループの頻度が 30 秒に設定されているため、動作グループの各動作は 30 秒ごとに再開されます。したがって、19 秒から 29 秒までの時間に開始する動作が存在しないため、18 秒の後に 10 秒のギャップが生じます。よって、動作グループ 2 の 2 番めの繰り返しは 30 秒に開始します。動作グループ 2 内の 10 個すべての動作は、設定された 20 秒のスケジュール期間内に均等に分散された間隔で開始しなければならないので、動作グループ 2 内の最後の動作（動作 10）は常に最初の動作（動作 1）の 18 秒後に開始します。

図に示すように、以下のようなイベントが発生します。

- 0 秒において、動作グループ 2 内の最初の動作（動作 1）が開始されます。

- 18 秒の時点で、動作グループ 2 の最後の動作（動作 10）が開始されます。つまり、動作グループ 1 の最初の繰り返し（スケジュール期間）がここで終了することを意味します。
- 19 ～ 29 秒に開始される動作はありません。
- 30 秒において、動作グループ 2 内の最初の動作（動作 1）が再び開始されます。動作グループ 2 の 2 番めの繰り返しがここから始まります。
- 48 秒において（2 番めの繰り返しが始まってから 18 秒後）、動作グループ 2 内の最後の動作（動作 10）が開始され、動作グループ 2 の 2 番めの繰り返しが終わります。
- 60 秒の時点で、動作グループ 2 の 3 番めの繰り返しが開始されます。

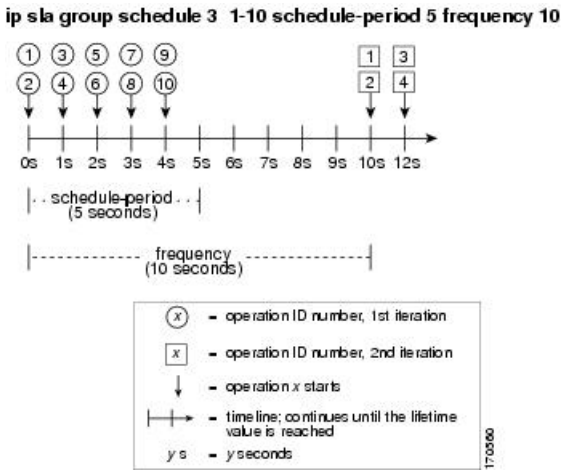
このプロセスは、動作グループ 2 のライフタイムが終わるまで続きます。ライフタイムの値は設定可能です。動作グループのデフォルトのライフタイムは無期限です。

IP SLA 動作の数がスケジュール期間よりも大きい場合の複数動作スケジューリング

グループ動作内の IP SLA 動作の開始の最小間隔は、1 秒です。そのため、スケジューリングされる動作の数がスケジュール期間よりも大きいと、IP SLA 複数動作スケジューリング機能は、同じ 1 秒間隔内で複数の動作が開始するようにスケジューリングします。スケジューリングされる動作の数を 1 秒間隔に均等に分割できない場合は、スケジュール期間の開始時に動作を均等に分割し、余った動作は最後の 1 秒の間隔で開始します。

次の図に、動作グループ 3 内の動作 1 から動作 10 のスケジューリングを示します。動作グループ 3 のスケジュール期間は 5 秒、頻度は 10 秒です。

図 *5 : IP SLA* 動作の数がスケジュール期間よりも大きい場合：均一な分配



この例では、スケジュール期間を動作の数で割ると、各 IP SLA 動作の開始時間が 1 秒未満になります（5 秒を 10 個の動作で割ると、0.5 秒毎に 1 動作になる）。グループ動作内の IP SLA 動作の開始の最小間隔は 1 秒なので、IP SLA 複数動作スケジューリング機能は、動作の数を

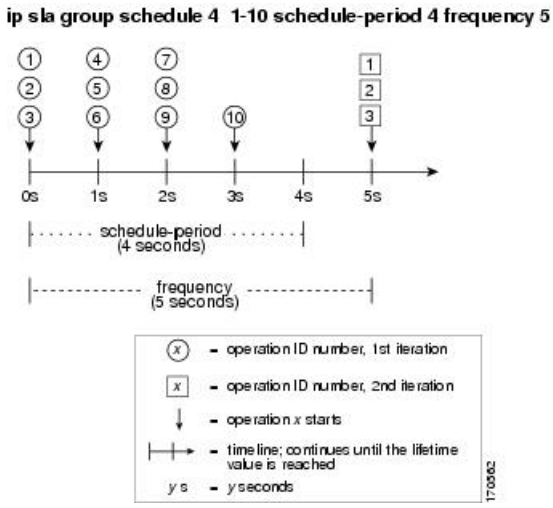
スケジュール期間で割ることにより（10 個の動作を 5 秒で割る）、各 1 秒間隔で開始しなければならない動作の数を代わりに計算します。そのため、前の図に示すように、1 秒おきに 2 つの動作が開始されます。

この例では頻度が 10 に設定されるので、動作グループ 3 の各繰り返しは、前の繰り返しの開始から 10 秒後に始まります。ただし、繰り返しの間に 5 秒の隙間があるため、この分散は最適なものではありません。

スケジューリングされる動作の数が 1 秒間隔に均等に分割されない場合は、スケジュール期間の開始時に動作が均等に分割され、余った動作は最後の 1 秒の間隔で開始します。

次の図に、動作グループ 4 内の動作 1 から動作 10 のスケジューリングを示します。動作グループ 4 のスケジュール期間は 4 秒、頻度は 5 秒です。

図 *6 : IP SLA* 動作の数がスケジュール期間よりも大きい場合：不均一な分配



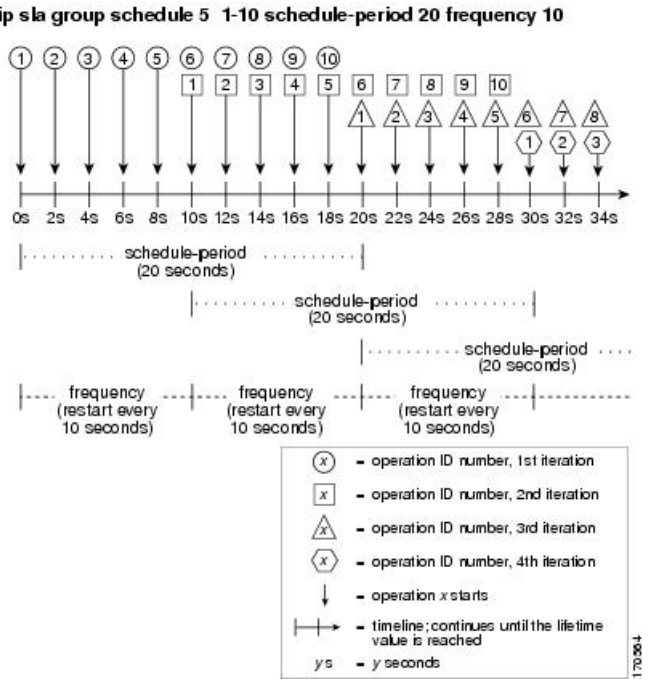
この例では、IP SLA 複数動作スケジューリング機能が、動作の数をスケジュール期間で割ることにより、各 1 秒間隔で開始しなければならない動作の数を計算します（10 個の動作が 4 秒で割られて、1 秒毎に 2.5 動作になる）。動作の数は 1 秒間隔では均等に分割できないため、この数は切り上げられ、残った動作は最後の 1 秒間隔に開始されることになります（図を参照）。

スケジュール期間が頻度よりも大きい場合の IP SLA 複数動作スケジューリング

頻度の値は、スケジュール グループが再開されるまでに経過する時間です。スケジュール期間が頻度よりも大きい場合は、動作グループのある繰り返し内の動作が、その後の繰り返しの動作と重なる期間ができます。

次の図に、動作グループ 5 内の動作 1 から動作 10 のスケジューリングを示します。動作グループ 5 のスケジュール期間は 20 秒、頻度は 10 秒です。

図 *7:* スケジュール期間が頻度よりも大きい場合の *IP SLA* グループ スケジューリング



この例では、動作グループ 5 内の最初の動作（動作 1）が 0 秒に開始します。動作グループ 5 内の 10 個すべての動作（動作 1 ～ 10）が、20 秒のスケジュール期間内に開始される必要があります。各 IP SLA 動作の開始時間は、スケジュール期間を動作の数で割ることにより（20 秒が 10 個の動作で割られる）、スケジュール期間にわたって均等に分散されます。したがって、各動作は前の動作の 2 秒後に開始されます。

動作グループ 5 の最初の繰り返しでは、動作 1 が 0 秒に開始し、動作 10（動作グループ内の最後の動作）は 18 秒に開始します。動作グループは 10 秒ごとに再開するように設定されているため（**frequency 10**）、動作グループ 5 の 2 番めの繰り返しは、最初の繰り返しの完了前である 10 秒に再び開始します。したがって、10 ～ 18 秒の期間中、最初の繰り返しの動作 6 ～ 10 が 2 番目の繰り返しの動作 1 ～ 5 と重なって実行されます（前の図を参照）。同様に、20 ～ 28 秒の期間中、2 番めの繰り返しの動作 6 ～ 10 は、3 番めの繰り返しの動作 1 ～ 5 と重なります。

この例では、動作 1 と動作 6 の開始時間は、同じ 2 秒の間隔内になりますが、厳密に同じ時間になる必要はありません。

動作の数をスケジュール期間よりも大きく設定することで、複数の動作が同じ 1 秒の間隔内で開始するように設定できるので、ここで説明されている設定は推奨されません。

IP SLA ランダム スケジューラ

IP SLA 複数動作スケジューリング機能を使用すると、複数の IP SLA 動作を、指定された期間にわたって均一に分散された間隔で開始し、指定された頻度で再開するようにスケジューリングできます。IP SLA ランダム スケジューラ機能を使用すると、複数の IP SLA 動作を、指定さ

れた期間にわたって均一に分散されたランダムな間隔で開始し、指定された頻度の範囲内に均一に分散されたランダムな頻度で再開するようにスケジューリングできるようになります。ランダム スケジューリングにより、ネットワーク パフォーマンスを評価するための統計的なメトリックが改善されます。

（注） IP SLA ランダム スケジューラ機能は、パケット間のランダム性が考慮されていないため、RFC2330 に準拠していません。

ランダム スケジューラ オプションは、デフォルトではディセーブルです。ランダム スケジューラ オプションをイネーブルにするには、グローバル コンフィギュレーション モードでグループ スケジュールを設定するときに、頻度範囲を設定する必要があります。動作のグループは、指定された頻度範囲の均一に分散されたランダムな頻度で再開されます。頻度の範囲を設定する場合は、次のガイドラインが適用されます。

- 頻度の範囲の開始値は、グループ動作のすべての動作のタイムアウト値よりも大きい値にする必要があります。
- 頻度の範囲の開始値は、スケジュール期間（グループ動作がスケジューリングされる時間）よりも大きい値にする必要があります。このガイドラインを順守することで、同じ動作が、スケジュール期間内に複数回スケジューリングされることがなくなります。

ランダム スケジューラ オプションがイネーブルである場合は、次のガイドラインが適用されます。

- グループ動作の個々の動作は、均一に分散されて、スケジュール期間にランダムな間隔で開始されます。
- 動作のグループは、指定された頻度範囲の均一に分散されたランダムな頻度で再開されます。
- グループ動作の各動作開始の最小間隔は、100 ミリ秒（0.1 秒）です。ランダム スケジューラ オプションがディセーブルの場合、最小間隔は 1 秒です。
- 特定の時間に開始されるようにスケジューリングできるのは、1 つの動作だけです。ランダム スケジューラ オプションがディセーブルの場合、複数の動作を同じ時間に開始できます。
- 最初の動作は常にスケジュール期間の 0 ミリ秒に開始されます。
- グループ動作の各動作が開始される順序はランダムです。

IP SLA 複数動作スケジューラの前提条件

- グループをスケジューリングする前に、IP SLA 動作をグループに含める設定を行う。
- 1 つのグループとしてスケジュールする IP SLA 動作を決定する。

- ネットワーク トラフィック タイプとネットワーク管理ステーションを特定する。
- ネットワークのトポロジおよびデバイスのタイプを特定する。
- 各動作に対するテストの頻度を決定する。

複数の IP SLA 動作のスケジューリング

ここでは、IP SLA 動作をスケジュールする方法について説明します。

始める前に

（注）
- 複数動作グループでスケジュールされたすべての動作の頻度が同じでなければなりません。
- 動作 ID 番号は、最大 125 文字までに制限されます。大きい整数値を動作 ID 番号に指定しないでください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period** *schedule-period-range* [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm*[*:ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh:mm:ss*}]
4. **exit**
5. **show ip sla group schedule**
6. **show ip sla configuration**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period** *schedule-period-range* [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life**{**forever** | *seconds*}] [**start-time** {*hh:mm*[*:ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after** *hh:mm:ss*}] 例： `switch(config)# ip sla group schedule 1 3,4,6-9`	スケジューリングされる IP SLA 動作グループ番号と動作番号の範囲をグローバル コンフィギュレーション モードで指定します。
ステップ 4	**exit** 例： `switch(config)# exit`	特権 EXEC モードに戻ります。
ステップ 5	**show ip sla group schedule** 例： `switch# show ip sla group schedule`	（任意）IP SLA グループ スケジュールの詳細を表示します。
ステップ 6	**show ip sla configuration** 例： `switch# show ip sla configuration`	（任意）IP SLA 設定の詳細を表示します。

IP SLA ランダム スケジューラのイネーブル化

ここでは、IP SLA ランダム スケジューラをイネーブルにする方法について説明します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period** *seconds* [**ageout** *seconds*] [**frequency** [*seconds*| **range** *random-frequency-range*]] [**life**{**forever** | *seconds*}] [**start-time**{*hh:mm*[*:ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after***hh:mm:ss*}]
4. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例：	特権 EXEC モードを有効にします。

	コマンドまたはアクション	目的
	`switch> enable`	プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period** *seconds* [**ageout** *seconds*] [**frequency** [*seconds*| **range** *random-frequency-range*]] [**life**{**forever** | *seconds*}] [**start-time**{*hh:mm*[*:ss*] [*monthday* | *daymonth*] | **pending** | **now** | **after***hh:mm:ss*}] 例： `switch(config)# ip sla group schedule 2 1-3 schedule-period 50 frequency range 80-100`	IP SLA 動作のグループのスケジューリング パラメータを指定します。 IP SLA ランダム スケジューラ オプションをイネーブルにするには、**frequency range** *random-frequency-range* キーワードおよび引数を設定する必要があります。
ステップ 4	**exit** 例： `switch(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

IP SLA 複数動作スケジューリングの確認

ここでは、IP SLA 複数動作スケジューリングを確認する方法について説明します。

手順の概要

1. **show ip sla statistics**
2. **show ip sla group schedule**
3. **show ip sla configuration**

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 ip sla statistics** 例： `switch# show ip sla statistics`	（任意）IP SLA 動作の詳細を表示します。
ステップ 2	**show ip sla group schedule** 例：	（任意）IP SLA グループ スケジュールの詳細を表示します。

	コマンドまたはアクション	目的
	`switch# show ip sla group schedule`	
ステップ 3	**show ip sla configuration** 例： `switch# show ip sla configuration`	（任意）IP SLA 設定の詳細を表示します。

例

複数の IP SLA 動作のスケジューリングが完了した後は、適切な **show** コマンドを使用して、最新の動作の詳細情報を確認できます。

次に、動作グループ 1 内の IP SLA 動作 1 ～ 20 を、60 秒のスケジュール期間と 1200 秒のライフ値でスケジュールする例を示します。デフォルトにより、頻度はスケジュール期間と同じです。この例では、開始間隔は 3 秒になります（スケジュール期間を動作の数で割った値）。

```
switch (config)# ip sla group schedule 1 1-20 schedule-period 60 life 1200
```

次に、スケジュールされた複数の IP SLA 動作の詳細を表示する例を示します。

```
switch# show ip sla group schedule
Group Entry Number: 1
Probes to be scheduled: 1-20
Total number of probes: 20
Schedule period: 60
Group operation frequency: Equals schedule period
Status of entry (SNMP RowStatus): Active
Next Scheduled Start Time: Start Time already passed
Life (seconds): 1200
Entry Ageout (seconds): never
```

次に、スケジュールされた複数の IP SLA 動作の詳細を表示する例を示します。この例では、IP SLA 動作が複数スケジュールされていること（TRUE）が示されています。

```
switch# show ip sla config 1
IP SLAs Infrastructure Engine-III
Entry number: 1
Owner:
Tag:
Operation timeout (milliseconds): 5000
Type of operation to perform: udp-jitter
Target address/Source address: 101.101.101.1/0.0.0.0
Target port/Source port: 5000/0
Type Of Service parameter: 0x0
Request size (ARR data portion): 32
Packet Interval (milliseconds)/Number of packets: 20/10
Verify data: No
Vrf Name: default
Control Packets: enabled
Schedule:
```

```
   Operation frequency (seconds): 60  (not considered if randomly scheduled)
   Next Scheduled Start Time: Start Time already passed
   Group Scheduled : TRUE
   Randomly Scheduled : FALSE
   Life (seconds): 3600
   Entry Ageout (seconds): never
   Recurring (Starting Everyday): FALSE
   Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Distribution Statistics:
   Number of statistic hours kept: 2
   Number of statistic distribution buckets kept: 1
   Statistic distribution interval (milliseconds): 20
```

次に、動作が等間隔でスケジュールされたときに、スケジュールされた複数の IP SLA 動作の最新の動作開始時間を表示する例を示します。

```
switch# show ip sla statistics | include Latest operation start time
Latest operation start time: *03:06:21.760 UTC Tue Oct 21 2003
Latest operation start time: *03:06:24.754 UTC Tue Oct 21 2003
Latest operation start time: *03:06:27.751 UTC Tue Oct 21 2003
Latest operation start time: *03:06:30.752 UTC Tue Oct 21 2003
Latest operation start time: *03:06:33.754 UTC Tue Oct 21 2003
Latest operation start time: *03:06:36.755 UTC Tue Oct 21 2003
Latest operation start time: *03:06:39.752 UTC Tue Oct 21 2003
Latest operation start time: *03:06:42.753 UTC Tue Oct 21 2003
Latest operation start time: *03:06:45.755 UTC Tue Oct 21 2003
Latest operation start time: *03:06:48.752 UTC Tue Oct 21 2003
Latest operation start time: *03:06:51.753 UTC Tue Oct 21 2003
Latest operation start time: *03:06:54.755 UTC Tue Oct 21 2003
Latest operation start time: *03:06:57.752 UTC Tue Oct 21 2003
Latest operation start time: *03:07:00.753 UTC Tue Oct 21 2003
Latest operation start time: *03:07:03.754 UTC Tue Oct 21 2003
Latest operation start time: *03:07:06.752 UTC Tue Oct 21 2003
Latest operation start time: *03:07:09.752 UTC Tue Oct 21 2003
Latest operation start time: *03:07:12.753 UTC Tue Oct 21 2003
Latest operation start time: *03:07:15.755 UTC Tue Oct 21 2003
Latest operation start time: *03:07:18.752 UTC Tue Oct 21 2003
```

複数の IP SLA 動作のスケジューリング構成例

以下に、20 秒のスケジュール期間で動作グループ 1 の IP SLA 動作 1 ～ 10 をスケジュールする例を示します。デフォルトにより、頻度はスケジュール期間と同じです。

```
switch# ip sla group schedule 1 1-10 schedule-period 20
```

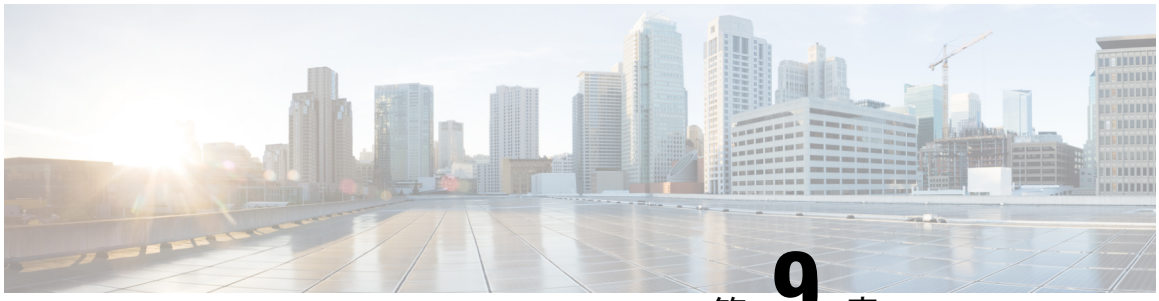
次に、スケジュールされた複数の IP SLA 動作の詳細を表示する例を示します。この例の最後の行では、IP SLA 動作が複数スケジューリングされていること（TRUE）が示されています。

```
switch# show ip sla group schedule
Multi-Scheduling Configuration:
Group Entry Number: 1
Probes to be scheduled: 1-10
Schedule period :20
Group operation frequency: 20
Multi-scheduled: TRUE
```

IP SLA ランダム スケジューラを有効にする構成例

次に、IP SLA 動作 1 ～ 3 をグループ（グループ 2 として指定）としてスケジューリングする例を示します。この例では、動作は、50 秒のスケジュール期間にわたって均一に分散されたランダムな間隔で開始するようにスケジューリングされます。最初の動作は、ただちに開始されるようにスケジューリングされます。間隔は、プローブが呼び出されるたびに、指定された範囲から毎回選択されます。ランダム スケジューラ オプションがイネーブルになり、動作のグループが再開する均一に分散されたランダムな頻度は、80 ～ 100 秒の範囲内で選択されます。

```
ip sla group schedule 2 1-3 schedule-period 50 frequency range 80-100 start-time now
```



第 9 章

IP SLA 動作の予防的しきい値モニタリングの設定

この章では、しきい値およびリアクション トリガーを使用した IP サービス レベル契約（SLA）の予防的モニタリング機能について説明します。

この章は、次の項で構成されています。

IP SLA リアクション構成に関する情報

IP SLA の反応は、モニタリング対象の値が指定のレベルを超えるか、下回った場合、または、タイムアウトや接続損失などのモニタリング対象のイベントが発生した場合にトリガーされるように設定します。IP SLA が測定するリアクション構成が高すぎたり、低すぎたりすると、IP SLA が、ネットワーク管理アプリケーションへの通知を生成したり、より多くのデータを収集する別の IP SLA 動作をトリガーしたりすることがあります。

IP SLA しきい値モニタリングおよび通知

IP SLA は、ほとんどの IP SLA 動作に関する平均ジッタ、単方向の遅延、双方向のラウンドトリップ時間（RTT）、および接続などのパフォーマンス パラメータについての予防的しきい値モニタリングおよび通知をサポートします。予防的モニタリング機能は、単方向ジッター、単方向のパケット損失、および単方向 VoIP 音声品質スコアリングを含む重要な VoIP 関連パラメータの反応しきい値を設定するためのオプションを提供します。

IP SLA の通知は、トリガーされた応答として設定されます。パケット損失、ジッター、平均動作スコア（MOS）統計情報は、IP SLA ジッター動作に固有です。通知はいずれかの方向（送信元から宛先、および宛先から送信元）の違反、またはパケット損失およびジッターの範囲外 RTT 値に対して生成できます。RTT 値が指定したしきい値を上回るか下回ると、トラップなどのイベントがトリガーされます。

応答条件が発生した場合、IP SLA ではシステム ロギング（syslog）メッセージを生成できます。システム ロギング メッセージは、CISCO-RTTMON-MIB を使用して簡易ネットワーク管理プロトコル（SNMP）トラップ（通知）として送信できます。IP SLA の SNMP トラップは、CISCO-RTTMON-MIB および CISCO-SYSLOG-MIB でサポートされます。

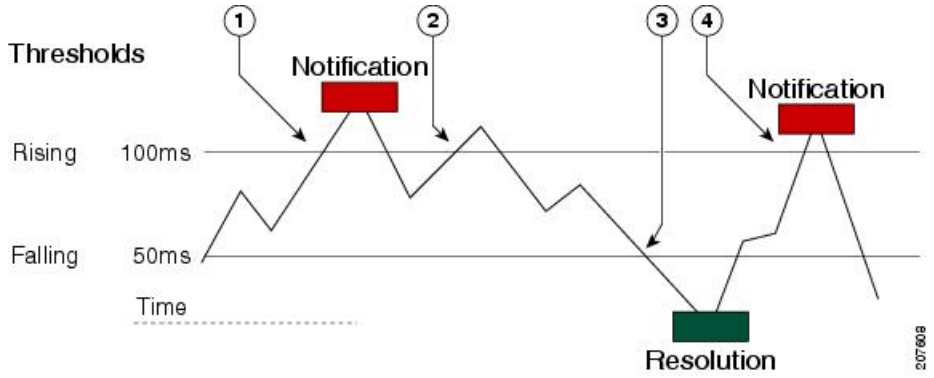
CISCO-SYSLOG-MIB のシビラティ（重大度）レベルは、SyslogSeverity INTEGER {emergency(1), alert(2), critical(3), error(4), warning(5), notice(6), info(7), debug(8)} です。

Cisco NX-OS ソフトウェアのシステム ロギング プロセスに対しては、異なるシビラティ（重大度）レベル値が定義されます。Cisco NX-OS ソフトウェアのシステム ロギング プロセスに対するシビラティ（重大度）レベルは、{emergency (0), alert (1), critical (2), error (3), warning (4), notice (5), informational (6), debugging (7)} です。

IP SLA しきい値違反は、Cisco システム ロギング プロセス内ではレベル 6（informational）としてロギングされますが、CISCO-SYSLOG-MIB からはレベル 7（info）トラップとして送信されます。

通知は、しきい値違反が発生するたびに発行されるわけではありません。次の図に、モニタリング対象要素が上限しきい値を超えたときに発生するトリガー リアクションの流れを示します。最初に上昇しきい値を超えたときに、イベントが送信され、通知が発行されます。後続のしきい値超過通知は、モニタリング対象の値が上昇しきい値を再び超える前に下限しきい値を下回った場合に限り発行されます。

図 *8 : IP SLA* のトリガーされた反応条件およびしきい値超過通知



1	最初に上昇しきい値を超えたときに、イベントが送信され、しきい値超過通知が発行されます。
2	上昇しきい値の超過違反が連続して発生しても、追加の通知は発行されません。
3	モニタリング対象の値が下限しきい値を下回っています。

4	上昇しきい値を超えたときに別のしきい値超過通知が発行されているのは、モニタリング対象の値が最初に下限しきい値を下回った後だけです。

（注） また、モニタリング対象の要素が下限しきい値を最初に下回った時点で（3）、下限しきい値超過通知が発行されます。下限しきい値超過違反に対する後続の通知が発行されるのは、上昇しきい値を超えた後で、モニタリング対象の値が下限しきい値を再び下回った場合に限られます。

ジッター動作に対する RTT 反応

ジッター動作に対する RTT 反応は、動作の最後にのみトリガーされます。これには、平均リターントリップ時間（RTTAvg）値とマッチングされる、リターントリップ時間の最新値（LatestRTT）が使用されます。

ジッター動作に対する RTT の SNMP トラップは、動作全体の平均リターントリップ時間（RTTAvg）値に基づいており、動作中に送信される個々のパケットの RTT 値は含まれません。たとえば、平均がしきい値を下回っている場合、実際には最大で半数のパケットがしきい値を上回っている可能性がありますが、あくまでも動作全体に対する値であるため、このような詳細は通知には含まれません。

RTTAvg しきい値違反に対しては、syslog メッセージだけがサポートされています。syslog メッセージは、CISCO-RTTMON-MIB から送信されます。

予防的しきい値モニタリングの設定

ここでは、トラップを生成したり、別の動作を開始するようにしきい値および反応トリガーを設定する方法について説明します。

始める前に

- 違反条件が満たされた場合に開始される IP SLA 動作を設定します。

（注）
- ジッター動作に対する RTT 反応は、動作の最後にのみトリガーされます。これには、リターントリップ時間の最新値（LatestRTT）が使用されます。
- ジッター動作に対する RTT の SNMP トラップは、動作全体に対するリターントリップ時間の平均値（RTTAvg）のみに基づいており、動作中に送信された個々のパケットのリターントリップ時間値は含まれません。RTTAvg しきい値違反に対しては、syslog メッセージだけがサポートされています。
- ジッター動作中の RTT 違反には、syslog メッセージだけがサポートされます。
- ジッター動作中以外の RTT 違反には、SNMP トラップだけがサポートされます。
- timeout、connectionLoss、または verifyError 以外の非 RTT 違反には、syslog メッセージのみがサポートされます。
- SNMP トラップと syslog メッセージの両方がサポートされているのは、timeout、connectionLoss、または verifyError 違反のみです。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip sla reaction-configuration** *operation-number* **react** *monitored-element* [**action-type** *option*] [**threshold-type** {**average** [*number-of-measurements*] | **consecutive** [*occurrences*] | **immediate** | **never** | **xofy** [*x-valuey-value*]}] [**threshold-value** *upper-thresholdlower-threshold*]
4. **ip sla reaction-trigger** *operation-number target-operation*
5. **ip sla logging traps**
6. **snmp-server enable traps ip sla**
7. **snmp-server host** {*hostname* | *ip-address*} [**vrf** *vrf-name*] [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*]
8. **exit**
9. **show ip sla reaction configuration** [*operation-number*]
10. **show ip sla reaction trigger** [*operation-number*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**ip sla reaction-configuration** *operation-number* **react** *monitored-element* [**action-type** *option*] [**threshold-type** {**average** [*number-of-measurements*] | **consecutive** [*occurrences*] | **immediate** | **never** | **xofy** [*x-valuey-value*]}] [**threshold-value** *upper-thresholdlower-threshold*] 例： `switch(config)# ip sla reaction-configuration 10 react jitterAvg threshold-type immediate threshold-value 5000 3000 action-type trapAndTrigger`	指定したしきい値違反に基づいて実行されるアクション（SNMP トラップまたは IP SLA トリガー）を設定します。
ステップ 4	**ip sla reaction-trigger** *operation-number target-operation* 例： `switch(config)# ip sla reaction-trigger 10 2`	（任意）違反条件が満たされた場合に、別の IP SLA 動作を開始します。 **ip sla reaction-configuration** コマンドが、**trapAndTrigger** または **triggerOnly** キーワードのいずれかを含めて構成された場合にのみ必要です。
ステップ 5	**ip sla logging traps** 例： `switch(config)# ip sla logging traps`	（任意）CISCO-RTTMON-MIB からの IP SLA syslog メッセージをイネーブルにします。
ステップ 6	**snmp-server enable traps ip sla** 例： `switch(config)# snmp-server enable traps ip sla`	（任意）システムによる CISCO-RTTMON-MIB トラップの生成をイネーブルにします。
ステップ 7	**snmp-server host** {*hostname* | *ip-address*} [**vrf** *vrf-name*] [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*] 例： `switch(config)# snmp-server host 10.1.1.1 public`	（任意）リモートホストにトラップを送信します。 **snmp-server enable traps** コマンドが構成されている場合にのみ必要です。
ステップ 8	**exit** 例： `switch(config)# exit`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 9	**show ip sla reaction configuration** [*operation-number*] 例： `switch# show ip sla reaction configuration 10`	（任意）予防的しきい値モニタリングの設定を表示します。
ステップ 10	**show ip sla reaction trigger** [*operation-number*] 例： `switch# show ip sla reaction trigger 2`	（任意）トリガーされるターゲット動作の設定ステータスおよび動作状態を表示します。

IP SLA 反応構成の設定例

MOS 値が 4.9（最高品質）を超えた時点、または 2.5（低品質）を下回った時点で SNMP ロギング トラップを送信するように、IP SLA 動作 10 を設定する例を示します。

```
switch(config)# ip sla reaction-configuration 10 react mos threshold-type immediate
threshold-value 490 250 action-type trapOnly
```

以下に、デフォルト設定を表示する例を示します。

```
switch# show ip sla reaction-configuration 1
Entry number: 1
Index: 1
Reaction: mos
Threshold Type: Immediate
Rising: 490
Falling: 250
Action Type: Trap only
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip sla reaction-configuration 10 react mos threshold-type immediate
threshold-value 490 250 action-type trapOnly
switch(config)# show ip sla reaction-configuration 1
Entry number: 1
Reaction: rtt
Threshold Type: Never
Rising (milliseconds): 5000
Falling (milliseconds): 3000
Threshold Count: 5
Threshold Count2: 5
Action Type: None
```

IP SLA リアクション構成の確認例

次の例では、出力内の [Reaction:] 値に示されているとおり、複数のモニタリング対象要素が IP SLA 動作 (1) に対して構成されています。

```
switch# show ip sla reaction-configuration

Entry Number: 1
Reaction: RTT
Threshold type: Never
Rising (milliseconds): 5000
Falling (milliseconds): 3000
Threshold Count: 5
Threshold Count2: 5
Action Type: None
Reaction: jitterDSAvg
Threshold type: average
Rising (milliseconds): 5
Falling (milliseconds): 3
Threshold Count: 5
Threshold Count2: 5
Action Type: triggerOnly
Reaction: jitterDSAvg
```

```
Threshold type:  immediate
Rising (milliseconds): 5
Falling (milliseconds): 3
Threshold Count: 5
Threshold Count2: 5
Action Type: trapOnly
Reaction: PacketLossSD
Threshold type:  immediate
Rising (milliseconds): 5
Threshold Falling (milliseconds): 3
Threshold Count: 5
Threshold Count2: 5
Action Type: trapOnly
```

SNMP 通知をトリガーするための構成例

次に、RTT または VoIP MOS のしきい値に違反した場合に、10.1.1.1 のリモート ホストに CISCO-SYSLOG-MIB トラップが送信されるように、予防的しきい値モニタリングを構成する例を示します。

```
! Configure the operation on source.
switch(config)# ip sla 1

switch(config-ip-sla)# udp-jitter 10.1.1.1 3000 codec g711alaw
switch(config-ip-sla-jitter)# exit

switch(config)# ip sla schedule 1 start now life forever

! Configure thresholds and reactions.
switch(config)# ip sla reaction-configuration 1 react rtt threshold-type immediate
threshold-value 3000 2000 action-type trapOnly

switch(config)# ip sla reaction-configuration 1 react MOS threshold-type consecutive 4
threshold-value 390 220 action-type trapOnly

switch(config)# ip sla logging traps

! The following command sends traps to the specified remote host.
switch(config)# snmp-server host 10.1.1.1 version 2c public

! The following command is needed for the system to generate CISCO-SYSLOG-MIB traps.
switch(config)# snmp-server enable traps
```

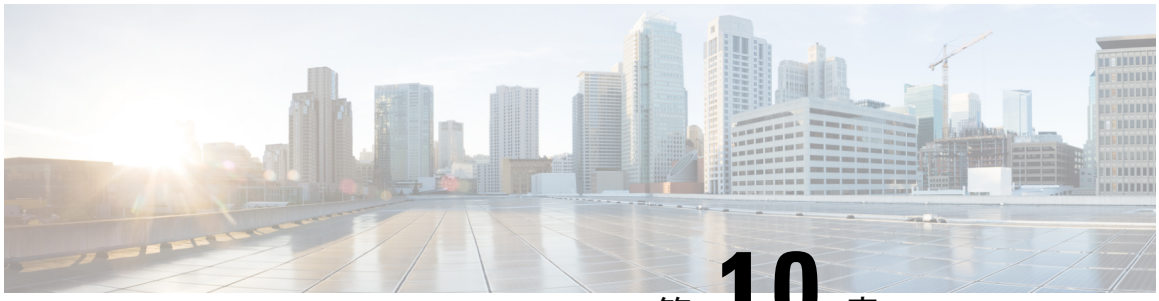
以下の例では、IP SLA しきい値違反通知が Cisco NX-OS システム ロギング プロセスでレベル 6 （informational）として生成されることが示されています。

```
3d18h:%RTT-6-SAATHRESHOLD:RTR(11):Threshold exceeded for MOS
```

以下の例では、同じ違反に対する CISCO-SYSLOG-MIB による SNMP 通知がレベル 7 （info）通知であることが示されています。

```
3d18h:SNMP:V2 Trap, reqid 2, errstat 0, erridx 0
 sysUpTime.0 = 32613038
```

```
snmpTrapOID.0 = ciscoSyslogMIB.2.0.1
clogHistoryEntry.2.71 = RTT
clogHistoryEntry.3.71 = 7
clogHistoryEntry.4.71 = SAATHRESHOLD
clogHistoryEntry.5.71 = RTR(11):Threshold exceeded for MOS
clogHistoryEntry.6.71 = 32613037
```



第 10 章

IPSLA オブジェクト トラッキングの構成

この章では、IP サービス レベル契約（SLA）の PBR オブジェクト トラッキング機能について説明します。

この章は、次の項で構成されています。

IP SLA PBR オブジェクト トラッキング

この機能により、ルートを使用する前にネクスト ホップが到達可能であることを確認できます。ネクスト ホップが到達可能でない場合、ポリシー ベース ルーティング（PBR）設定で定義されている別のルートが使用されます。ルートマップに他のルートがない場合は、ルーティング テーブルが使用されます。

オブジェクト トラッキング

オブジェクト トラッキングでは、次のようなオブジェクトがモニタされます。

- インターフェイスの回線プロトコルの状態
- ルーティング テーブル内のエントリの存在

PBR などのクライアントは、特定のトラッキング対象オブジェクトを登録し、それらのオブジェクトの状態が変化した時点でアクションを実行することができます。

IP SLA PBR オブジェクト トラッキングの概要

PBR オブジェクト トラッキング機能により、トラッキング プロセスで使用できるすべてのオブジェクトへのポリシー ベース ルーティング（PBR）アクセスが可能になります。トラッキング プロセスを使って、ICMP ping 到達可能性、ルーティング隣接関係、リモート デバイス上

で実行中のアプリケーション、Routing Information Base（RIB）内のルートなどの個々のオブジェクトや、インターフェイス回線プロトコルの状態をトラッキングできます。

オブジェクトトラッキングが機能する仕組み：PBRがトラッキングプロセスに特定のオブジェクトを追跡するように通知すると、そのオブジェクトで変更が発生した時点で、トラッキングプロセスがPBRに通知します。

IP SLA PBR オブジェクト トラッキングの構成

手順の概要

1. **configure terminal**
2. **ip sla** *operation-number*
3. **icmp-echo** *destination-ip-address*
4. **exit**
5. **ip sla schedule** *operation-number* **life forever start-time now**
6. **track** *object-number* **ip sla** *entry-number* **reachability**
7. **exit**
8. **ip access-list standard** *access-list-name*
9. **permit ip** *source destination*
10. **ipv6 access-list** *access-list-name*
11. **permit ipv6** *source destination*
12. **exit**
13. **route-map** *map-tag*
14. **match ip address** *access-list-name*
15. **match ipv6 address** *access-list-name*
16. **set ip next-hop verify-availability** *next-hop-address* **track** *object*
17. **set ipv6 next-hop verify-availability** *next-hop-address* **track** *object*
18. **exit**
19. **interface** *type number*
20. **ip address** *ip-address mask*
21. **ipv6 address** *ip-address mask*
22. **ip policy route-map** *map-tag*
23. **ipv6 policy route-map** *map-tag*
24. **end**
25. **show track** *object-number*
26. **show route-map** *map-name*

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 terminal** 例：	グローバル設定モードを開始します。

	コマンドまたはアクション	目的
	`switch# configure terminal`	
ステップ 2	**ip sla** *operation-number* 例： `switch(config)# ip sla 1`	Cisco IOS IP サービス レベル契約（SLA）の動作設定を開始し、IP SLA 構成モードを開始します。
ステップ 3	**icmp-echo** *destination-ip-address* 例： `switch(config-ip-sla)# icmp-echo 10.3.3.2`	IP SLA Internet Control Message Protocol（ICMP）エコー プローブ動作を設定します。
ステップ 4	**exit** 例： `switch(config-ip-sla)# exit`	IP SLA モニタ構成モードを終了し、ルータをグローバル構成モードに戻します。
ステップ 5	**ip sla schedule** *operation-number* **life forever start-time now** 例： `switch(config)# ip sla schedule 1 life forever start-time now`	単一の Cisco IOS IP SLA 動作のスケジューリング パラメータを設定します。 • この例では、IP SLA 動作の時間パラメータを設定します。 （注） 他の IP SLA 動作を構成およびスケジュールするには、ステップ 2 から 5 を繰り返します。
ステップ 6	**track** *object-number* **ip sla** *entry-number* **reachability** 例： `switch(config)# track 1 ip sla 1 reachability`	オブジェクトの到達可能性を追跡し、トラッキング構成モードを開始します。 （注） 他の動作を追跡するには、この手順を繰り返します。
ステップ 7	**exit** 例： `switch(config-track)# exit`	トラッキング コンフィギュレーション モードを終了し、ルータをグローバル コンフィギュレーション モードに戻します。
ステップ 8	**ip access-list standard** *access-list-name* 例： `switch(config)# ip access-list standard ACL`	パケットのフィルタリングを有効にするために、IP アクセス リストのアクセス制御リスト（ACL）を定義します。
ステップ 9	**permit ip** *source destination* 例：	条件を満たすトラフィックを許可する、IP アクセス制御リスト（ACL）のルールを作成します。

	コマンドまたはアクション	目的
	`switch(config-acl)# permit ip 192.0.2.0/24 198.51.100.0/24`	
ステップ 10	**ipv6 access-list** *access-list-name* 例： `switch(config)# ipv6 access-list IPv6ACL`	パケットのフィルタリングを有効にするために、IPv6 アクセス リスト ACL を定義します。
ステップ 11	**permit ipv6** *source destination* 例： `switch(config-ipv6-acl)# permit ipv6 2001:DB8::/32 2001:DB8::/48`	条件を満たすトラフィックを許可する、IP アクセス制御リスト（ACL）のルールを作成します。
ステップ 12	**exit** 例： `switch(config-ipv6-acl)# exit`	ルータ構成モードを終了し、グローバル構成モードに戻ります。
ステップ 13	**route-map** *map-tag* 例： `switch(config)# route-map PBR`	ルート マップを指定し、ルート マップ コンフィギュレーション モードを開始します。
ステップ 14	**match ip address** *access-list-name* 例： `switch(config-route-map)# match ip address ACL`	標準アクセス リストで許可された宛先 IPv4 ネットワーク番号アドレスを含むルートがあれば、配布します。
ステップ 15	**match ipv6 address** *access-list-name* 例： `switch(config-route-map)# match ipv6 address IPv6ACL`	標準アクセス リストで許可された宛先 IPv6 ネットワーク番号アドレスを含むルートがあれば、配布します。
ステップ 16	**set ip next-hop verify-availability** *next-hop-address* **track** *object* 例： `switch(config-route-map)# set ip next-hop verify-availability 198.51.100.2 track 1`	ルートマップを設定し、トラッキング対象オブジェクトの到達可能性を確認します。 （注） この手順を繰り返して、他のトラッキング対象オブジェクトの到達可能性を確認するためのルート マップを設定します。
ステップ 17	**set ipv6 next-hop verify-availability** *next-hop-address* **track** *object* 例：	ルートマップを設定し、トラッキング対象オブジェクトの到達可能性を確認します。

	コマンドまたはアクション	目的
	`switch(config-route-map)# set ipv6 next-hop verify-availability 2001:DB8:1::1 track 1`	（注） この手順を繰り返して、他のトラッキング対象オブジェクトの到達可能性を確認するためのルート マップを設定します。
ステップ 18	**exit** 例： `switch(config-route-map)# exit`	ルートマップ構成モードを終了し、ルータをグローバル構成モードに戻します。
ステップ 19	**interface** *type number* 例： `switch(config)# interface ethernet 0/0`	インターフェイスのタイプと番号を指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 20	**ip address** *ip-address mask* 例： `switch(config-if)# ip address 10.2.2.1 255.255.255.0`	インターフェイスのプライマリ IP アドレスを指定します。
ステップ 21	**ipv6 address** *ip-address mask* 例： `switch(config-if)# ipv6 address 2001:DB8::/48`	インターフェイスのプライマリ IPv6 アドレスを指定します。
ステップ 22	**ip policy route-map** *map-tag* 例： `switch(config-if)# ip policy route-map PBR`	ポリシールーティングをイネーブルにし、ポリシールーティングに使用するルート マップを指定します。
ステップ 23	**ipv6 policy route-map** *map-tag* 例： `switch(config-if)# ipv6 policy route-map PBR`	IPv6 ポリシー ルーティングを有効にし、ポリシールーティングに使用するルート マップを指定します。
ステップ 24	**end** 例： `switch(config-if)# end`	インターフェイス コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 25	**show track** *object-number* 例： `switch# show track 1`	（任意）トラッキング情報を表示します。 このコマンドを使用して、設定を確認します。

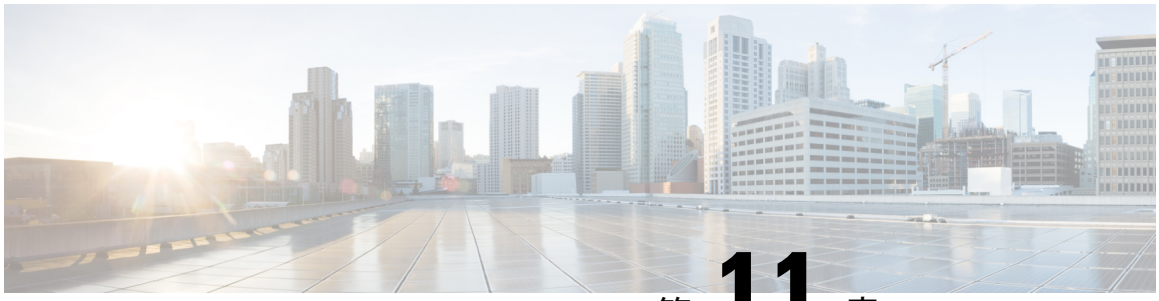
	コマンドまたはアクション	目的
ステップ 26	**show route-map** *map-name* 例： `switch# show route-map PBR`	（任意）ルート マップ情報を表示します。

例：IP SLA PBR オブジェクト トラッキングの構成

以下に、PBR に対して構成されたオブジェクト トラッキングの例を示します。

```
! Configure and schedule IP SLA operations
ip sla 1
 icmp-echo 10.3.3.2
ip sla schedule 1 life forever start-time now
!
ip sla 2
 udp-echo 10.4.4.2
ip sla schedule 2 life forever start-time now
!
ip sla 3
 icmp-echo 10.5.5.2
ip sla schedule 3 life forever start-time now
!
ip sla 4
 icmp-echo 10.6.6.2
ip sla schedule 4 life forever start-time now
!
ip sla 5
 icmp-echo 10.7.7.2
ip sla schedule 5 life forever start-time now
!
! Configure Object Tracking to track the operations
!
track 1 ip sla 1 reachability
track 2 ip sla 2 reachability
track 3 ip sla 3 reachability
track 4 ip sla 4 reachability
track 5 ip sla 5 reachability
!
! Configure ACL
ip access-list standard ACL
 permit ip 10.2.2.0/24 10.1.1.1/32
!
! Configure PBR policing on the router
route-map PBR
 match ip address ACL
 set ip next-hop verify-availability 10.3.3.2 track 1
 set ip next-hop verify-availability 10.4.4.2 track 2
 set ip next-hop verify-availability 10.5.5.2 track 3
!
! Apply PBR policy on the incoming interface of the router.
interface ethernet 0/0
 ip address 10.2.2.1 255.255.255.0
  ip policy route-map PBR
!
! Display PBR related information
show route-map
```

```
show track brief
show ip sla stat
show ip sla application
!
```



第 11 章

IP SLA DNS 動作の設定

この章では、IP サービス レベル契約（SLA）の DNS 動作機能について説明します。

この章は、次の項で構成されています。

IP SLA DNS 動作

ここでは、DNS要求を送信するのに要する時間と応答を受信するのに要する時間の差異を測定するために IP SLA DNS 動作を設定する方法について説明します。

IP SLA DNS 動作に関する注意事項と制約事項

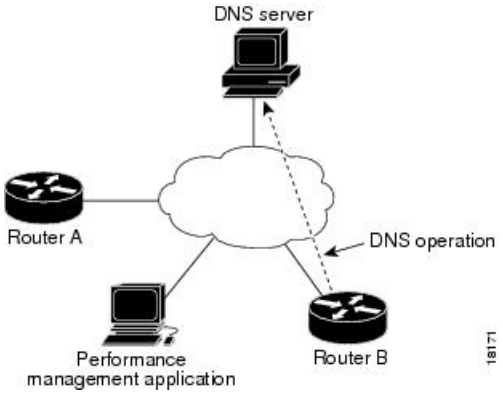
- キーワードが付いている**show**コマンド**internal**はサポートされていません。
- IP SLA DNS 動作では、IPv6 はサポートされていません。

DNS の動作

DNS 動作では、DNS 要求を送信するのに要する時間と、応答を受信するのに要する時間の差異を測定します。DNS は、ネットワーク ノードの名前をアドレスに変換するためにインターネットで使用されます。IP SLA DNS 動作は、ホスト名を指定した場合は IP アドレスを問い合わせ、IP アドレスを指定した場合はホスト名を問い合わせます。

以下の図では、デバイス B を送信元 IP SLA デバイスとし、接続先デバイスを DNS サーバーとする DNS 動作が設定されています。

図 *9 : DNS* 動作



要求を DNS サーバーに送信するのに要する時間とデバイス B が応答を受信するのに要する時間の差異を測定することにより、接続応答時間が算出されます。得られた DNS ルックアップ時間は、DNS のパフォーマンスの分析に役立ちます。DNS ルックアップ時間が短いと、Web サーバー アクセスが高速になります。

送信元デバイスでの基本 DNS 動作の設定

手順の概要

1. **configure terminal**
2. **feature sla sender**
3. **ip sla** *operation-number*
4. **dns** {*destination-ip-address* | *destination-hostname*} **name-server** *ip-address* [**source-ip** {*ip-address* | *hostname*} **source-port** *port-number*]
5. **frequency** *seconds*
6. **end**

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 terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。

	コマンドまたはアクション	目的
ステップ 3	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 4	**dns** {*destination-ip-address* | *destination-hostname*} **name-server** *ip-address* [**source-ip** {*ip-address* | *hostname*} **source-port** *port-number*] 例： `switch(config-ip-sla)# dns host1 name-server 172.20.2.132`	DNS 動作を定義し、IP SLA DNS コンフィギュレーション モードを開始します。
ステップ 5	**frequency** *seconds* 例： `switch(config-ip-sla-dns)# frequency 60`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 6	**end** 例： `switch(config-ip-sla-dns)# end`	特権 EXEC モードに戻ります。

送信元デバイスでのオプション パラメータを使用した DNS 動作の設定

手順の概要

1. **configure terminal**
2. **feature sla sender**
3. **ip sla** *operation-number*
4. **dns** {*destination-ip-address* | *destination-hostname*} **name-server** *ip-address* [**source-ip** {*ip-address* | *hostname*} **source-port** *port-number*]
5. **history buckets-kept** *size*
6. **history distributions-of-statistics-kept** *size*
7. **history filter**{**none** | **all** | **overThreshold** | **failures**}
8. **frequency** *seconds*
9. **history hours-of-statistics-kept** *hours*
10. **history lives-kept** *lives*
11. **owner** *owner-id*
12. **history statistics-distribution-interval** *milliseconds*
13. **tag** *text*

14. **threshold** *milliseconds*
15. **timeout** *milliseconds*
16. **end**

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 terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 3	**ip sla** *operation-number* 例： `switch(config)# ip sla 10`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 4	**dns** {*destination-ip-address* | *destination-hostname*} **name-server** *ip-address* [**source-ip** {*ip-address* | *hostname*} **source-port** *port-number*] 例： `switch(config-ip-sla)# dns host1 name-server 172.20.2.132`	DNS 動作を定義し、IP SLA DNS コンフィギュレーション モードを開始します。
ステップ 5	**history buckets-kept** *size* 例： `switch(config-ip-sla-dns)# history buckets-kept 25`	（任意）IP SLA 動作のライフタイム中に保持する履歴バケット数を設定します。
ステップ 6	**history distributions-of-statistics-kept** *size* 例： `switch(config-ip-sla-dns)# history distributions-of-statistics-kept 5`	（任意）IP SLA 動作中にホップ単位で保持する統計情報の配信数を設定します。
ステップ 7	**history filter**{**none** | **all** | **overThreshold** | **failures**} 例： `switch(config-ip-sla-dns)# history filter failures`	（任意）IP SLA 動作の履歴テーブルに格納する情報のタイプを定義します。

	コマンドまたはアクション	目的
ステップ 8	**frequency** *seconds* 例： `switch(config-ip-sla-dns)# frequency 30`	（任意）指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 9	**history hours-of-statistics-kept** *hours* 例： `switch(config-ip-sla-dns)# history hours-of-statistics-kept 4`	（任意）IP SLA 動作の統計情報を保持する時間数を設定します。
ステップ 10	**history lives-kept** *lives* 例： `switch(config-ip-sla-dns)# history lives-kept 2`	（任意）IP SLA 動作の履歴テーブルに格納するライフ数を設定します。
ステップ 11	**owner** *owner-id* 例： `switch(config-ip-sla-dns)# owner admin`	（任意）IP SLA 動作の簡易ネットワーク管理プロトコル（SNMP）所有者を設定します。
ステップ 12	**history statistics-distribution-interval** *milliseconds* 例： `switch(config-ip-sla-dns)# history statistics-distribution-interval 10`	（任意）IP SLA 動作で維持する各統計情報の配信間隔を設定します。
ステップ 13	**tag** *text* 例： `switch(config-ip-sla-dns)# tag TelnetPollServer1`	（任意）IP SLA 動作のユーザー指定 ID を作成します。
ステップ 14	**threshold** *milliseconds* 例： `switch(config-ip-sla-dns)# threshold 9000`	（任意）IP SLA 動作によって作成されるネットワーク モニタリング統計情報を計算するための上限しきい値を設定します。
ステップ 15	**timeout** *milliseconds* 例： `switch(config-ip-sla-dns)# timeout 10000`	（任意）IP SLA 動作がその要求パケットからの応答を待機する時間を設定します。
ステップ 16	**end** 例： `switch(config-ip-sla-dns)# end`	特権 EXEC モードに戻ります。

IP SLA 動作のスケジューリング

（注）
- スケジュールされるすべての IP SLA 動作がすでに構成されている必要があります。
- 複数動作グループでスケジュールされたすべての動作の頻度が同じでなければなりません。
- 複数動作グループに追加する動作 ID 番号のリストは、カンマ（,）を含めて最大 125 文字に制限する必要があります。

手順の概要

1. **configure terminal**
2. 次のいずれかを使用します。
   - **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {[*hh*:*mm*:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *seconds*] [**recurring**]
   - **ip sla group schedule** *group-operation-number operation-id-numbers* {**schedule-period** *schedule-period-range* | **schedule-together**} [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life** {**forever** | *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh*:*mm*[:*ss*]}]
3. **exit**
4. **show ip sla group schedule**
5. **show ip sla configuration**

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 terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	次のいずれかを使用します。 • **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {[*hh*:*mm*:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *seconds*] [**recurring**] • **ip sla group schedule** *group-operation-number operation-id-numbers* {**schedule-period** *schedule-period-range* | **schedule-together**} [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life**	個々の IP SLA 動作のスケジューリング パラメータを設定します。 複数動作スケジューラ用に IP SLA 動作グループ番号と動作番号の範囲を指定します。

	コマンドまたはアクション	目的

	{**forever** | *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh*:*mm*[:*ss*]}]
	例：
	`switch(config)# ip sla schedule 10 life forever start-time now`
	例：
	`switch(config)# ip sla group schedule 1 3,4,6-9 life forever start-time now`	

ステップ 3	**exit** 例： `switch(config)# exit`	特権 EXEC モードに戻ります。
ステップ 4	**show ip sla group schedule** 例： `switch# show ip sla group schedule`	（任意）IP SLA グループ スケジュールの詳細を表示します。
ステップ 5	**show ip sla configuration** 例： `switch# show ip sla configuration`	（任意）IP SLA 設定の詳細を表示します。

DNS 動作の設定例

ここでは「DNS 動作」の項の図「DNS 動作」に示されているように、デバイス B から DNS サーバー（IP アドレス 172.20.2.132）への DNS 動作を設定する例を示します。動作は、ただちに開始されるようにスケジューリングされます。この例では、ターゲットアドレスはホスト名であり、DNS 動作はホスト名 host1 に関連付けられた IP アドレスを DNS サーバーに問い合わせます。DNS サーバーでの設定は必要ありません。

```
feature sla sender
ip sla 11
 dns host1 name-server 172.20.2.132
 frequency 50
 timeout 8000
 tag DNS-Test
ip sla schedule 11 start-time now
```

送信元デバイスでの基本 DNS 動作の設定例

以下に、送信元デバイスでの基本 DNS 動作の設定例を示します。

```
switch# configure terminal
switch(config)# feature sla sender
switch(config)# ip sla 10
switch(config-ip-sla)# dns host1 name-server 172.20.2.132
switch(config-ip-sla-dns)# frequency 60
switch(config-ip-sla-dns)# end
```

送信元デバイスでのオプション パラメータを使用した DNS 動作の設定例

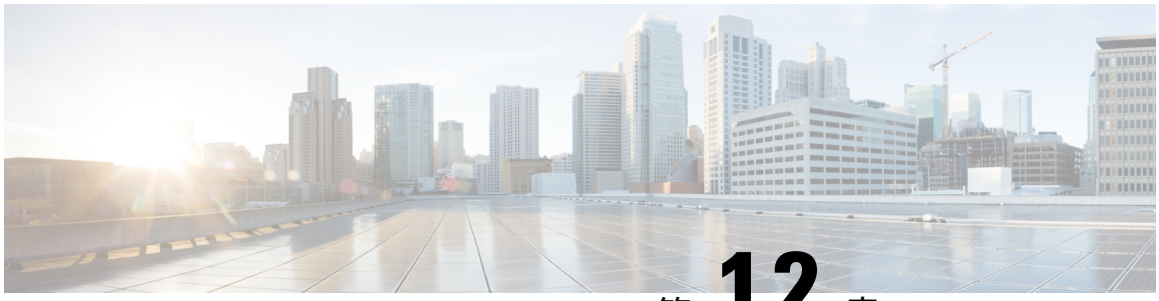
以下に、送信元デバイスで最適なパラメータを使用してDNS動作を設定する例を示します。

```
switch# configure terminal
switch(config)# feature sla sender
switch(config-ip-sla)# dns host1 name-server 172.20.2.132
switch(config)# ip sla 10
switch(config-ip-sla)# dns host1 name-server 172.20.2.132
switch(config-ip-sla-dns)# history buckets-kept 25
switch(config-ip-sla-dns)# history distributions-of-statistics-kept 5
switch(config-ip-sla-dns)# history filter failures
switch(config-ip-sla-dns)# frequency 30
switch(config-ip-sla-dns)# history hours-of-statistics-kept 4
switch(config-ip-sla-dns)# history lives-kept 2
switch(config-ip-sla-dns)# owner admin
switch(config-ip-sla-dns)# history statistics-distribution-interval 10
switch(config-ip-sla-dns)# tag TelnetPollServer1
switch(config-ip-sla-dns)# threshold 9000
switch(config-ip-sla-dns)# timeout 10000
switch(config-ip-sla-dns)# end
```

IP SLA 動作のスケジューリングの構成例

以下に、IP SLA 動作をスケジュールする例を示します。

```
switch# configure terminal
switch(config)# feature sla sender
switch(config)# ip sla schedule 10 life forever start-time now
switch(config)# exit
switch# show ip sla group schedule
switch# show ip sla configuration
```



第 12 章

IP SLA ICMP エコー動作の設定

このモジュールでは、IPv4 を使用する 2 台のデバイスの間のエンドツーエンド応答時間をモニタするように、IP サービス レベル契約（SLA）インターネット制御メッセージ プロトコル（ICMP）エコー動作を設定する方法について説明します。

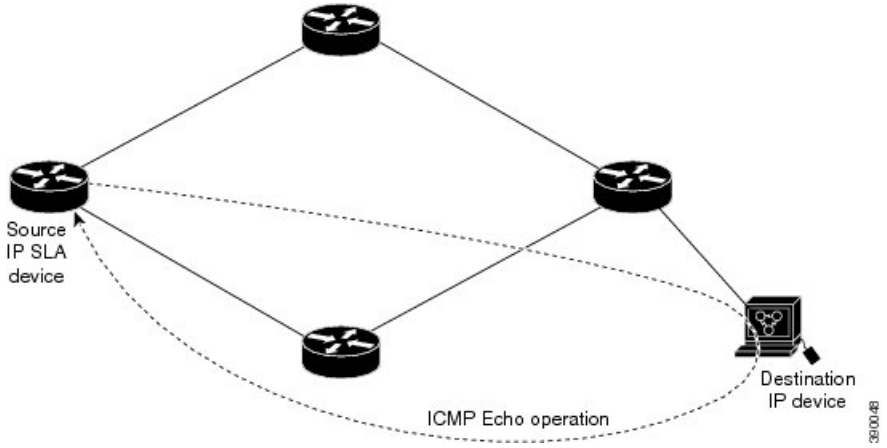
この章は、次の項で構成されています。

ICMP エコー動作

Internet Control Message Protocol（ICMP）エコー動作は、IPv4 または IPv6 を使用する 2 台のデバイス間のエンドツーエンド応答時間を測定します。応答時間は、ICMP エコー要求メッセージを宛先に送信して ICMP エコー応答を受信するまでの時間を測定して算出します。ICMP エコーは、ネットワーク接続問題のトラブルシューティングに役立ちます。ICMP エコー動作の結果を表示および分析することで、ネットワーク IP 接続の実況状況を判断できます。

次の図では、ICMP エコー動作は ping ベースのプローブを使用して送信元 IP SLA デバイスと宛先 IP デバイスの間の応答時間を測定します。多くのお客様が、応答時間の測定に IP SLA ICMP ベース動作、社内 ping テスト、または ping ベース専用プローブを使用しています。

図 *10 : ICMP* エコー動作



IP SLA ICMP エコー動作と ICMP ping テストは同じ IETF 仕様に準拠しているので、どちらの方法でも同じ応答時間が得られます。

IP SLA ICMP エコー動作に関する注意事項と制限事項

- キーワードが付いている**show**コマンド**internal**はサポートされていません。
- 宛先デバイスには Cisco のネットワーキング デバイスを使用することを推奨しますが、RFC 862 エコー プロトコルをサポートするネットワーキング デバイスであれば使用できます。

ICMP エコー動作の設定

（注） 接続先デバイスで IP SLA Responder を構成する必要はありません。

次のいずれかの操作を行います。

- 送信元デバイスでの基本 ICMP エコー動作の構成
- オプション パラメータを使用した ICMP エコー動作の構成

送信元デバイスでの基本 ICMP エコー動作の構成

手順の概要

1. **configure terminal**
2. **feature sla sender**

3. **ip sla** *operation-number*
4. **icmp-echo** {*destination-ipv4-address* | | *destination-hostname*} [**source-ip** {*ipv4-address* | | *hostname*} | **source-interface** *interface-name*]
5. **end**

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 terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 3	**ip sla** *operation-number* 例： `switch(config)# ip sla 6`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 4	**icmp-echo** {*destination-ipv4-address*		*destination-hostname*} [**source-ip** {*ipv4-address*		*hostname*}	**source-interface** *interface-name*] 例： `switch(config-ip-sla)# icmp-echo 192.0.2.134` 例：	ICMP エコー動作を定義し、IP SLA ICMP エコー コンフィギュレーション モードを開始します。
ステップ 5	**end** 例： `switch(config-ip-sla-echo)# end`	IP SLA エコー構成モードを終了し、特権 EXEC モードに戻ります。

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、IP サービス レベル契約（SLA）動作に予防的しきい値条件と反応トリガーを追加するには、「IP SLA 動作の予防的しきい値モニタリングの構成」の章の「予防的しきい値モニタリングの構成」の項を参照してください。

オプション パラメータを使用した ICMP エコー動作の設定

始める前に

このタスクは、送信元デバイスで実行します。

手順の概要

1. **configure terminal**
2. **feature sla sender**
3. **ip sla** *operation-number*
4. **icmp-echo** {*destination-ipv4-address* | | *destination-hostname*} [**source-ip** {*ipv4-address* | | *hostname*} | **source-interface** *interface-name*]
5. （任意） **history buckets-kept** *size*
6. （任意） **history distributions-of-statistics-kept** *size*
7. （任意） **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets*
8. （任意） **history filter** {**none** | **all** | **overThreshold** | **failures**}
9. （任意） **frequency** *seconds*
10. （任意） **history hours-of-statistics-kept** *hours*
11. （任意） **history lives-kept** *lives*
12. （任意） **owner** *owner-id*
13. （任意） **request-data-size** *bytes*
14. （任意） **history statistics-distribution-interval** *milliseconds*
15. （任意） **tag** *text*
16. （任意） **threshold** *milliseconds*
17. （任意） **timeout** *milliseconds*
18. （任意） {**tos** | }*number*
19. （任意） **verify-data**
20. （任意） **vrf** {*vrf-name* | **default** | **management**}
21. **end**

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 terminal** 例： `switch# configure terminal`	グローバル設定モードを開始します。
ステップ 2	**feature sla sender** 例： `switch(config)# feature sla sender`	IP SLA 動作機能を有効にします。
ステップ 3	**ip sla** *operation-number* 例： `switch(config)# ip sla 6`	IP SLA 動作の設定を開始し、IP SLA コンフィギュレーション モードに移行します。
ステップ 4	**icmp-echo** {*destination-ipv4-address*		*destination-hostname*} [**source-ip** {*ipv4-address*		*hostname*}	**source-interface** *interface-name*] 例：	エコー動作を定義し、IP SLA エコー コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`switch(config-ip-sla)# icmp-echo 192.0.2.134 source-ip 192.0.2.132`	
ステップ 5	（任意） **history buckets-kept** *size* 例： `switch(config-ip-sla-echo)# history buckets-kept 25`	IP SLA 動作のライフタイム中に保持する履歴バケット数を設定します。
ステップ 6	（任意） **history distributions-of-statistics-kept** *size* 例： `switch(config-ip-sla-echo)# history distributions-of-statistics-kept 5`	IP SLA 動作中にホップ単位で保持する統計情報の配信数を設定します。
ステップ 7	（任意） **history enhanced** [**interval** *seconds*] [**buckets** *number-of-buckets* 例： `switch(config-ip-sla-echo)# history enhanced interval 900 buckets 100`	IP SLA 動作に対する拡張履歴収集を有効にします。
ステップ 8	（任意） **history filter** {**none** | **all** | **overThreshold** | **failures**} 例： `switch(config-ip-sla-echo)# history filter failures`	IP SLA 動作の履歴テーブルに格納する情報のタイプを定義します。
ステップ 9	（任意） **frequency** *seconds* 例： `switch(config-ip-sla-echo)# frequency 30`	指定した IP SLA 動作を繰り返す間隔を設定します。
ステップ 10	（任意） **history hours-of-statistics-kept** *hours* 例： `switch(config-ip-sla-echo)# history hours-of-statistics-kept 4`	IP SLA 動作の統計情報を保持する時間数を設定します。
ステップ 11	（任意） **history lives-kept** *lives* 例： `switch(config-ip-sla-echo)# history lives-kept 5`	IP SLA 動作の履歴テーブルに格納するライフ数を設定します。
ステップ 12	（任意） **owner** *owner-id* 例： `switch(config-ip-sla-echo)# owner admin`	IP SLA 動作の簡易ネットワーク管理プロトコル（SNMP）所有者を設定します。
ステップ 13	（任意） **request-data-size** *bytes* 例：	IP SLA 動作の要求パケットのペイロードにおけるプロトコル データ サイズを設定します。

	コマンドまたはアクション	目的
	`switch(config-ip-sla-echo)# request-data-size 64`	
ステップ 14	（任意） **history statistics-distribution-interval** *milliseconds* 例： `switch(config-ip-sla-echo)# history statistics-distribution-interval 10`	IP SLA 動作で維持する各統計情報の配信間隔を設定します。
ステップ 15	（任意） **tag** *text* 例： `switch(config-ip-sla-echo)# tag TelnetPollServer1`	IP SLA 動作のユーザー指定 ID を作成します。
ステップ 16	（任意） **threshold** *milliseconds* 例： `switch(config-ip-sla-echo)# threshold 10000`	IP SLA 動作によって作成されるネットワーク モニタリング統計情報を計算するための上限しきい値を設定します。
ステップ 17	（任意） **timeout** *milliseconds* 例： `switch(config-ip-sla-echo)# timeout 10000`	IP SLA 動作がその要求パケットからの応答を待機する時間を設定します。
ステップ 18	（任意） {**tos** | }*number* 例： `switch(config-ip-sla-echo)# tos 160`	IPv4 ネットワークに限り、IP SLA 動作の IPv4 ヘッダーのタイプ オブ サービス（ToS）バイトを定義します。 IPv6 ネットワークに限り、サポートされている IP SLA 動作に対する、IP SLA 動作の IPv6 ヘッダーのトラフィック クラス バイトを定義します。
ステップ 19	（任意） **verify-data** 例： `switch(config-ip-sla-echo)# verify-data`	IP SLA 動作が各応答パケットに対してデータ破壊の有無をチェックするようにします。
ステップ 20	（任意） **vrf** {*vrf-name* | **default** | **management**} 例： `switch(config-ip-sla-echo)# vrf vpn-A`	IP SLA 動作を使用して、マルチプロトコル ラベル スイッチング（MPLS）バーチャル プライベート ネットワーク（VPN）内をモニタリングできるようにします。
ステップ 21	**end** 例： `switch(config-ip-sla-echo)# end`	IP SLA エコー構成モードを終了し、特権 EXEC モードに戻ります。

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、IPサービス レベル契約（SLA）動作に予防的しきい値条件と反応トリガーを追加するには、「IP SLA 動作の予防的しきい値モニタリングの構成」の章の「予防的しきい値モニタリングの構成」の項を参照してください。

IP SLA 動作のスケジューリング

ここでは、IP SLA 動作をスケジュールする方法について説明します。

始める前に

（注）
- スケジュールされるすべての IP SLA 動作がすでに設定されている必要があります。
- 複数動作グループでスケジュールされたすべての動作の頻度が同じでなければなりません。
- 複数動作グループに追加される 1 つ以上の動作 ID 番号のリストは、カンマ（,）を含めて最大 125 文字に制限されます。

ヒント
- IP SLA 動作が実行されておらず、統計を生成していない場合は、 **verify-data** コマンドを動作の構成に追加して（IP SLA 構成モードで設定）、データ検証を有効にします。イネーブルになると、各動作の応答が破損していないかどうかがチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- **debug ip sla trace** コマンドを使用し、 および **debug ip sla error** コマンドは、IP SLA 動作に関する問題のトラブルシューティングを行うためのコマンドです。

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかを実行します。
   - **ip sla schedule** *operation-number* [**life forever** { | *seconds*}] [**starttime** {*hh : mm*[: *ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh : mm : ss*}] [**ageout** *seconds*] [**recurring**]

     例：

     ```
     ip sla schedule operation-number [life {forever | seconds}] [starttime {hh :
     mm[: ss] [month day | day month] | pending | now | after hh : mm : ss}] [ageout
      seconds] [recurring]
     ```

- **ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period** *schedule-period-range* [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life**{**forever** | *seconds*}] [**starttime**{ *hh:mm*[:*ss*] [*month day* | *day month]* | **pending** | **now** | **after** *hh:mm:ss*}]

  例：

  ```
  switch(config)# ip sla group schedule 1 3,4,6-9
  ```

4. **exit**
5. **show ip sla group schedule**
6. **show ip sla configuration**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。

ステップ 3	次のいずれかを実行します。
	• **ip sla schedule** *operation-number* [**life forever** { | *seconds*}] [**starttime** {*hh : mm*[: *ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh : mm : ss*}] [**ageout** *seconds*] [**recurring**]
	例：
	`ip sla schedule operation-number [life {forever | seconds}] [starttime {hh : mm[: ss] [month day | day month] | pending | now | after hh : mm : ss}] [ageout seconds] [recurring]`
	• **ip sla group schedule** *group-operation-number operation-id-numbers* **schedule-period** *schedule-period-range* [**ageout** *seconds*] [**frequency** *group-operation-frequency*] [**life**{**forever** | *seconds*}] [**starttime**{ *hh:mm*[:*ss*] [*month day* | *day month]* | **pending** | **now** | **after** *hh:mm:ss*}]
	例：
	`switch(config)# ip sla group schedule 1 3,4,6-9`
		-
		• 個々の IP SLA 動作の場合のみ：
		個々の IP SLA 動作のスケジューリング パラメータを設定します。
		• 複数動作スケジューラの場合のみ：
		スケジューリングされる IP SLA 動作グループ番号と動作番号の範囲をグローバル コンフィギュレーション モードで指定します。

ステップ 4	**exit** 例： `switch(config)# exit`	特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
ステップ 5	**show ip sla group schedule** 例： `switch# show ip sla group schedule`	（任意）IP SLA グループ スケジュールの詳細を表示します。
ステップ 6	**show ip sla configuration** 例： `switch# show ip sla configuration`	（任意）IP SLA 設定の詳細を表示します。

次のタスク

トラップを生成する目的、または別の動作を開始する目的で、動作に予防的しきい値条件と反応トリガーを追加するには、「予防的しきい値モニタリングの設定」の項を参照してください。

IP SLA 動作の結果を表示し、内容を確認するには、**show ip sla statistics** コマンドを使用します。 を実行する前に、ユーザ名がフィギュレーション ファイルに指定されていることを確認してください。サービス レベル契約の基準に対応するフィールドの出力を確認すると、サービス メトリックが許容範囲内であるかどうかを判断する役に立ちます。

トラブルシューティングのヒント

- IP SLA 動作が実行されておらず、統計を生成していない場合は、 **verify-data** コマンドを動作の構成に追加して（IP SLA 構成モードで設定）、データ検証を有効にします。データ検証をイネーブルにすると、各動作の応答で破損の有無がチェックされます。通常の動作時に **verify-data** コマンドを使用すると、不要なオーバーヘッドがかかるので注意してください。
- **debug ip sla trace** および **debug ip sla error** コマンドは、IP SLA 動作に関する問題のトラブルシューティングを行うためのコマンドです。

次の作業

トラップを生成する目的、または別の動作を開始する目的で、IP サービス レベル契約（SLA）動作に予防的しきい値条件と反応トリガーを追加するには、「IP SLA 動作の予防的しきい値モニタリングの構成」の章の「予防的しきい値モニタリングの構成」の項を参照してください。

IP SLA ICMP エコー動作の設定例

例：送信元デバイスでの基本 ICMP エコー動作の構成

以下に、送信元デバイスでの基本 ICMP エコー動作を構成する例を示します。

```
switch# configure terminal
switch(config)# feature sla sender
switch(config)# ip sla 6
switch(config-ip-sla)# icmp-echo 192.0.2.134 source-ip 192.0.2.132
switch(config-ip-sla-echo)# end
```

例：オプション パラメータを使用した ICMP エコー動作の構成

次に、IPv4を使用した IP SLA ICMP エコー動作の構成例を示します。動作はただちに開始され、無期限に実行されます。

```
switch# configure terminal
switch(config)# feature sla sender
switch(config)# ip sla 6
switch(config-ip-sla)# icmp-echo 192.0.2.134 source-ip 192.0.2.132
switch(config-ip-sla-echo)# frequency 300
switch(config-ip-sla-echo)# request-data-size 38
switch(config-ip-sla-echo)# tos 160
switch(config-ip-sla-echo)# timeout 6000
switch(config-ip-sla-echo)# tag SFO-RO
switch(config-ip-sla-echo)# end
```

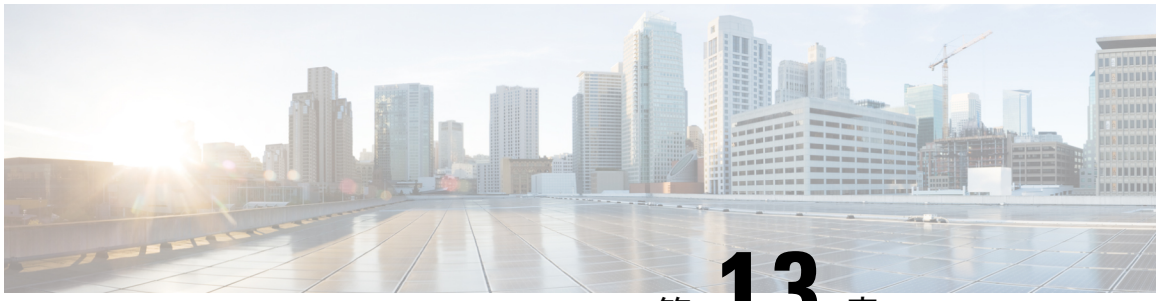
次に、IPv6を使用した IP SLA ICMP エコー動作の構成例を示します。動作はただちに開始され、無期限に実行されます。

```
switch# configure terminal
switch(config)# feature sla sender
switch(config)# ip sla 6
switch(config-ip-sla)# icmp-echo 2016:1:1::2 source-ip 2016:1:1::1
switch(config-ip-sla-echo)# frequency 300
switch(config-ip-sla-echo)# request-data-size 38
switch(config-ip-sla-echo)# traffic-class 160
switch(config-ip-sla-echo)# timeout 6000
switch(config-ip-sla-echo)# tag SFO-RO
switch(config-ip-sla-echo)# end
```

例：IP SLA 動作のスケジューリング

次に、すでに構成されている IP SLA 動作をスケジュールする例を示します。

```
switch# configure terminal
switch(config)# feature sla sender
switch(config)# ip sla schedule 6 life forever start-time now
switch(config)# exit
```



第 13 章

IP SLA TWAMP Responder

このモジュールでは、ネットワーク上のシスコ デバイスとシスコ以外の TWAMP 制御デバイス間の IP パフォーマンスを測定するために、シスコ デバイスで IETF Two-Way Active Measurement Protocol（TWAMP）Responder を設定する方法について説明します。

IP SLA TWAMP Responder の前提条件

IP SLA TWAMP Responder が機能するには、TWAMP 制御クライアントとセッション送信元をネットワークに設定する必要があります。

IP SLA TWAMP Responder の制限事項

- IP SLA TWAMP Responder v1.0 では、TWAMP サーバーとセッション リフレクタは、同一のシスコ デバイスに設定する必要があります。
- TWAMP クライアントおよびセッション送信側はサポートされていません。
- 1 つの TWAMP Responder に対して最大 10 の制御セッションを構成し、確立できます。
- TWAMP 光モードはサポートされていません。

IP SLA TWAMP Responder に関する情報

TWAMP

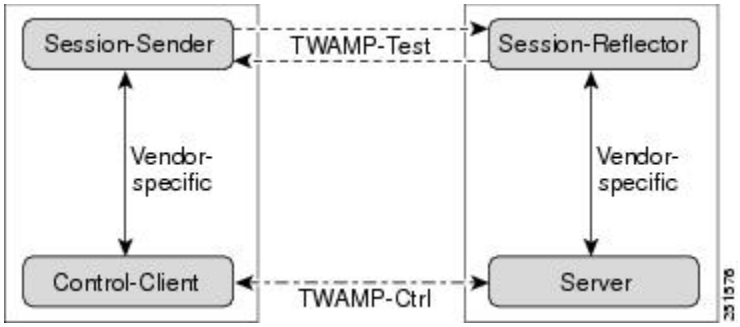
IETF Two-Way Active Measurement Protocol（TWAMP）は、TWAMP プロトコルをサポートする 2 つのデバイス間でのラウンドトリップ ネットワーク パフォーマンスの測定に関する規格を定めたものです。TWAMP 制御プロトコルは、パフォーマンス測定セッションを設定するために使用されます。TWAMP テストプロトコルは、パフォーマンス測定プローブを送受信するために使用されます。

TWAMP アーキテクチャは、モニタリング セッションの開始とパケットの交換に関与する次の 4 つの論理エンティティで構成されます。

- 制御クライアントは、TWAMP テストセッションをセットアップし、開始および停止を行います。
- セッション送信元は、セッション リフレクタに送信される TWAMP テスト パケットをインスタンス化します。
- セッション リフレクタは、TWAMP テスト パケットの受信時に、測定パケットを反映します。セッション リフレクタは、TWAMP 内のパケット統計情報を収集しません。
- TWAMP サーバーは、1 つ以上の TWAMP セッションを管理するエンド システムで、エンドポイント内のセッションごとのポートを設定することもできます。サーバーは TCP ポート 135 でリッスンします。セッション リフレクタとサーバーは、IP SLA 動作で TWAMP Responder を構成します。

TWAMP は柔軟性の異なるエンティティを定義しますが、単一デバイスでロールの論理的なマージも可能にし、実装が容易になります。次の図に、TWAMP アーキテクチャを構成する 4 つのエンティティを示します。

図 *11 : TWAMP* のアーキテクチャ

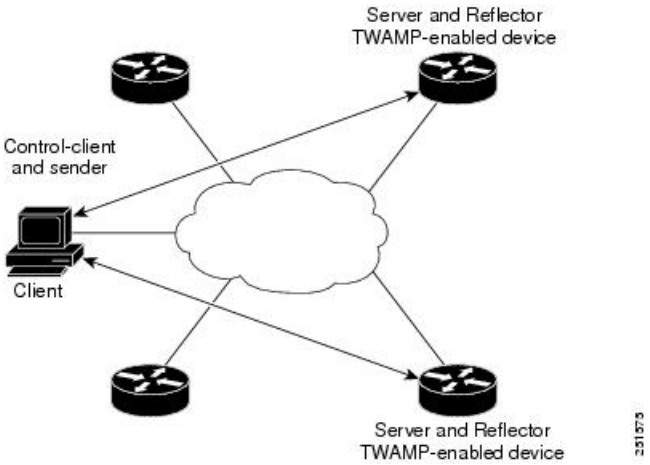


IP SLA TWAMP Responder v1.0

TWAMP Responder は、TWAMP をサポートする別のデバイスでコントロール クライアントおよびセッション送信元と相互運用します。IP SLA TWAMP Responder v1.0 機能では、Responder を構成するセッション リフレクタおよび TWAMP サーバーは、同じデバイス上に設置する必要があります。

次の図では、1 つのデバイスがコントロール クライアントおよびセッション送信元（TWAMP 制御デバイス）で、他の 2 つのデバイスが IP SLA TWAMP Responder として構成された Cisco デバイスです。各 IP SLA TWAMP Responder は、TWAMP サーバーおよびセッション リフレクタの両方として機能します。

図 *12 :* 基本的な *TWAMP* 展開での *IP SLA TWAMP Responder*



IP SLA TWAMP Responder の設定方法

TWAMP サーバーの設定

（注） IP SLA TWAMP Responder v1.0 では、TWAMP サーバーとセッション リフレクタは、同一のデバイスに設定されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **feature sla twamp-server**
4. **ip sla server twamp**
5. **port** *port-number*
6. **timer inactivity** *seconds*

7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla twamp-server** 例： `switch(config)# feature sla twamp-server`	デバイスを TWAMP サーバーとして構成します。
ステップ 4	**ip sla server twamp** 例： `switch(config)# ip sla server twamp`	TWAMP サーバー構成モードを開始します。
ステップ 5	**port** *port-number* 例： `switch(config-twamp-srvr)# port 9000`	（任意）TWAMP サーバーが接続および制御要求を受信するために使用するポートを設定します。
ステップ 6	**timer inactivity** *seconds* 例： `switch(config-twamp-srvr)# timer inactivity 300`	（任意）TWAMP 制御セッションの非アクティビティ タイマーを設定します。
ステップ 7	**end** 例： `switch(config-twamp-srvr)# end`	特権 EXEC モードに戻ります。

セッション リフレクタの設定

（注） IP SLA TWAMP Responder v1.0 では、TWAMP サーバーとセッション リフレクタは、同一のデバイスに設定されます。

手順の概要

1. **enable**

2. **configure terminal**
3. **feature sla responder**
4. **ip sla responder twamp**
5. **timeout** *seconds*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `switch> enable`	特権 EXEC モードを有効にします。 • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	**configure terminal** 例： `switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**feature sla responder** 例： `switch(config)# feature sla responder`	デバイスを TWAMP サーバーとして構成します。
ステップ 4	**ip sla responder twamp** 例： `switch(config)# ip sla server twamp`	TWAMP レスポンダ構成モードを開始します。
ステップ 5	**timeout** *seconds* 例： `switch(config-twamp-ref)# timeout 300`	（任意）TWAMP テスト セッションのタイムアウトを設定します。
ステップ 6	**end** 例： `switch(config-twamp-ref)# end`	特権 EXEC モードに戻ります。

IP SLA TWAMP レスポンダの設定例

IP SLA TWAMP Responder v1.0 の例

次の例と部分的な出力は、同一のシスコ デバイスで IP SLA TWAMP Responder v1.0 用の TWAMP サーバーとセッション リフレクタを設定する方法を示します。この設定では、ポート 862 は TWAMP サーバーが接続および制御要求を受信するために使用する

（デフォルト）ポートです。サーバー リスナーのデフォルト ポートは、RFC 指定のポートで、必要に応じて再設定できます。

（注） IP SLA TWAMP Responder が機能するには、制御クライアントとセッション送信元をネットワークに設定する必要があります。

```
switch> enable
switch# configure terminal
switch(config)# ip sla server twamp
switch(config-twamp-srvr)# exit
switch(config)# ip sla responder twamp
switch(config-twamp-ref)# end
switch> show running-config
.
.
.
ip sla responder
ip sla responder twamp
ip sla server twamp
```

IP SLA TWAMP Responder 設定の確認

IP SLA TWAMP Responder の設定情報を表示するには、次のいずれかのタスクを実行します。

コマンド	目的
show ip sla twamp standards	IP SLA TWAMP レスポンダで使用されている RFC 標準を表示します。
show ip sla twamp session	IP SLA TWAMP セッションに関する送信側と受信側の情報を表示します。
show ip sla twamp connection [**detail**|**requests**]	IP SLA TWAMP 接続に関する情報を表示します。次のオプションを指定できます。 • **Details**：現在の接続の詳細を表示します。詳細には、クライアント IP アドレス、クライアント ポート、VRF、モード、接続状態、制御状態、およびテスト要求の数が含まれます。 • **Requests**：現在の接続リクエストを表示します。

次の例は、IP SLA TWAMP Responder で使用されている現在の RFC 標準を示しています。

```
switch# show ip sla twamp standards
Feature                   Organization      Standard
TWAMP                     Server IETF       RFC5357
TWAMP                     Reflector IETF    RFC5357
```

次の例は、IP SLA TWAMP セッションに関する送信側と受信側の情報を示しています。

```
switch# show ip sla twamp session
IP SLAs Responder TWAMP is: Enabled
Recvr Addr: 30.30.30.1
Recvr Port: 7147
Sender Addr: 30.30.30.2
Sender Port: 50790
Sender VRF: default
Session Id: 30.30.30.1:15918249420668138422:DF55BEE9
Connection Id: 21
```

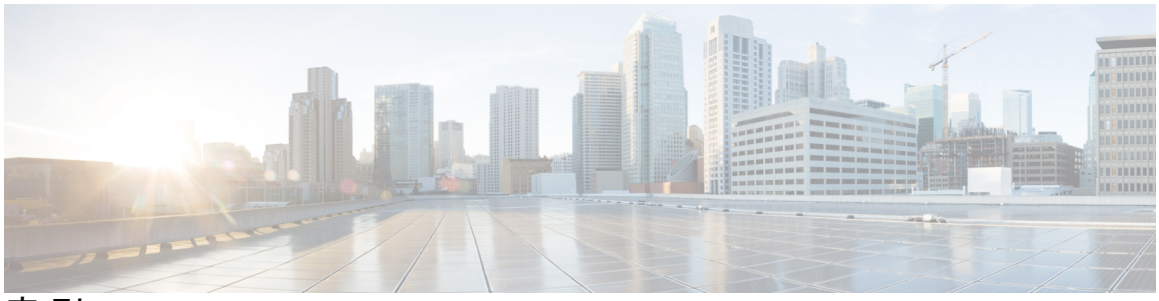
次の例では、現在のクライアント接続の詳細を表示しています。

```
switch# show ip sla twamp connection detail
Connection Id:          21
  Client IP Address:    30.30.30.2
  Client Port:          58316
  Client VRF:           default
  Mode:                 Unauthenticated
  Connection State:     Connected
  Control State:        Active
  Number of Test Requests - 0:1
```

その他の参考資料

標準および RFC

標準/RFC	タイトル
RFC 5357	『*Two-Way Active Measurement Protocol (TWAMP)*』
RFC 4656	『*One-way Active Measurement Protocol (OWAMP)*』



索 引

T

U

V

い

こ

し

た

て

ひ

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。