

# SNMP を使った OSPF 構成管理

## 内容

### [概要](#)

[OSPF の背景](#)

[プロセス定義](#)

[プロセス オーナー](#)

[プロセスの目的](#)

[プロセス パフォーマンス インジケータ](#)

[プロセスの入力](#)

[プロセスの出力](#)

[タスク定義](#)

[初期化タスク](#)

[反復タスク](#)

[データの識別](#)

[一般的なデータの特徴](#)

[SNMP データの識別](#)

[RMON データの識別](#)

[Syslog データの識別](#)

[Cisco IOS CLI データの識別](#)

[データ収集](#)

[SNMP データ収集](#)

[RMON データの収集](#)

[Syslog データの収集](#)

[Cisco IOS CLI データの収集](#)

[データ表示](#)

[OSPF エリア レポート](#)

[OSPF インターフェイス レポート](#)

[OSPF 隣接ルータ レポート](#)

[商用および一般的なインターネット監視ツール](#)

[SNMP ポーリング データ](#)

[データ収集アルゴリズムの例](#)

[関連情報](#)

## 概要

Open Shortest Path First ( OSPF ) ルーティング プロトコルは、RFC 2328 OSPF Version 2 で定義されています。この文書の目的は、組織が OSPF の設計計画に照らし合わせて OSPF の展開を検証するために構成管理の手順を実行し、また、意図した設計との長期的な一貫性が保証されるようにするために OSPF の展開を定期的に監査できるようにする、手順の枠組みについて説明することです。

この文書では、ITU-T によって定義された FCAPS ( fault, configuration, accounting/inventory, performance, security ) モデルの構成管理機能に重点を置いています。構成管理は、ITU-T M.3400 で定義され、NE ( ネットワーク要素 ) の管理、識別、データ収集、およびデータ提供などの機能を提供します。

この文書の情報は、次に示すいくつかの主要なセクションに分けて説明されています。

「OSPF バックグラウンド」のセクションでは、OSPF の展開における重要な点など、OSPF のバックグラウンドに関する情報の技術的な概要について説明しています。

「プロセス定義」のセクションでは、OSPF 構成管理を行うために使用するプロセス定義の概要について説明します。プロセスの詳細は、目標、パフォーマンス インジケータ、入力、出力、および個々のタスクの観点から説明されています。

「[タスク定義](#)」セクションでは、プロセス タスク定義について詳細に説明しています。個々のタスクについては、目的、タスクの入力、タスクの出力、タスクを実行するために必要なリソース、およびタスクを実装するために必要な職務上の技能の観点から説明しています。

「データの識別」のセクションでは、OSPF のためのデータの識別について説明しています。データの識別では、情報の発信元または情報の所在地を判断します。たとえば、情報は、Simple Network Management Protocol ( SNMP; 簡易ネットワーク管理プロトコル ) の Management Information Base ( MIB; 管理情報ベース )、Syslog によって生成された ログ ファイル、または Command Line Interface ( CLI; コマンドライン インターフェイス ) からのみアクセス可能な内部データ構造などの中にシステムによって構築されます。

「データ収集」のセクションでは、OSPF データの収集について説明しています。データ収集は、データの場所と密接な関係があります。たとえば、SNMP MIB データは、トラップ、Remote Monitoring ( RMON; リモート モニタリング ) のアラームやイベント、またはポーリングなど、いくつかのメカニズムで収集されます。内部データ構造によって保持されているデータは、自動スクリプトか、あるいはユーザがそのシステムに手動でログインして CLI コマンドを発行し、その出力を記録するという方法によって収集されます。

「データの表示」セクションでは、データがレポート形式で表現される方法の例について説明します。それらのデータは識別され、収集された後、解析されます。この文書では、OSPF 構成データの記録や比較に使用される可能性のあるレポートの例を紹介します。

「商用および一般的なインターネット監視ツール」、「SNMP ポーリング データ」、および「データ収集アルゴリズムの例」の各セクションでは、OSPF 構成管理の手順を実装するためのツールの開発に関する情報について説明しています。

## [OSPF の背景](#)

OSPF は、単一の自律システムで使用されるように設計された内部ゲートウェイ プロトコルです。OSPF では、リンクステートまたは shortest-path first ( SPF; 最短パス優先 ) ベースのテクノロジーを使用しています。これに対し、Routing Information Protocol ( RIP; ルーティング情報プロトコル ) などのルーティング プロトコルでは、ディスタンスベクターまたはベルマンフォードのテクノロジーが使用されています。個々の link-state advertisement ( LSA; リンクステート アドバタイズメント ) では、自律システムの全体など、OSPF ルーティング ドメインの一部について説明しています。このような LSA はルーティング ドメイン全体にフラッドされ、リンクステート データベースを形成します。ドメイン内の各ルータでは、全く同一のリンクステート データベースを保持します。リンクステート データベースの同期は、信頼性の高いフラッディング アルゴリズムによって維持されています。各ルータでは、このリンクステート データベースから最短経

路のツリーを計算し、計算を行っているルータ自身をツリーのルートとしてルーティングテーブルを構築します。この計算は、通常はダイクストラのアルゴリズムと呼ばれます。

LSA は小さなもので、各 LSA は OSPF ルーティングドメインの小さな一部分についての説明をします。具体的には、1 台のルータの近隣情報、1 つのトランジットネットワークの近隣情報、1 つのエリア間経路、または 1 つの外部経路などです。

次の表では、OSPF の主要な機能が説明されています。

機能	説明
隣接関係	OSPF ルータのペアが隣接関係になったとき、この 2 つのルータではデータベースのサマリーを OSPF データベース交換パケットの形式で交換し、それぞれのリンクステートデータベースを同期させます。その後、隣接関係ルータは、信頼性の高いフラッディングアルゴリズムを使用して、それぞれのリンクステートデータベースの同期を維持します。シリアル回線で接続されているルータは、常に隣接関係にあります。マルチアクセスネットワーク（イーサネット）では、そのネットワークに接続されているルータはすべて、designated router（DR; 代表ルータ）および backup designated router（BDR; バックアップ代表ルータ）の両方と隣接関係を持ちます。
代表ルータ	すべてのマルチアクセスネットワークにおいて 1 台の代表ルータが選ばれている場合、ネットワークのローカル環境を表すネットワーク LSA がその代表ルータから発信されます。また、代表ルータはフラッディングアルゴリズムにおいても特別な役割を持っています。フラッディング処理では、ネットワーク上のすべてのルータが代表ルータに対して LSA を送受信することにより、リンクステートデータベースを同期させています。
バックアップ代表ルータ	現在の DR が動作しなくなった場合には、マルチアクセスネットワーク上で BDR が選ばれ、DR の移行を速やかに行います。BDR が処理を引き継いだ場合、その local-area network（LAN; ローカルエリアネットワーク）で隣接関係処理を行う必要はありません。また、DR の消失が通知される以前での DR 不在の状態でも、BDR により信頼性の高いフラッディングアルゴリズムが続行できます。
非ブロードキャストマルチアクセスネット	OSPF では、フレームリレーの public data network（PDN; 公衆データ網）などのネットワークを LAN のように扱います。ただし、これらのネットワークに接続しているルータが初めて相互に認識するためには、構成情報の追加が必要です。

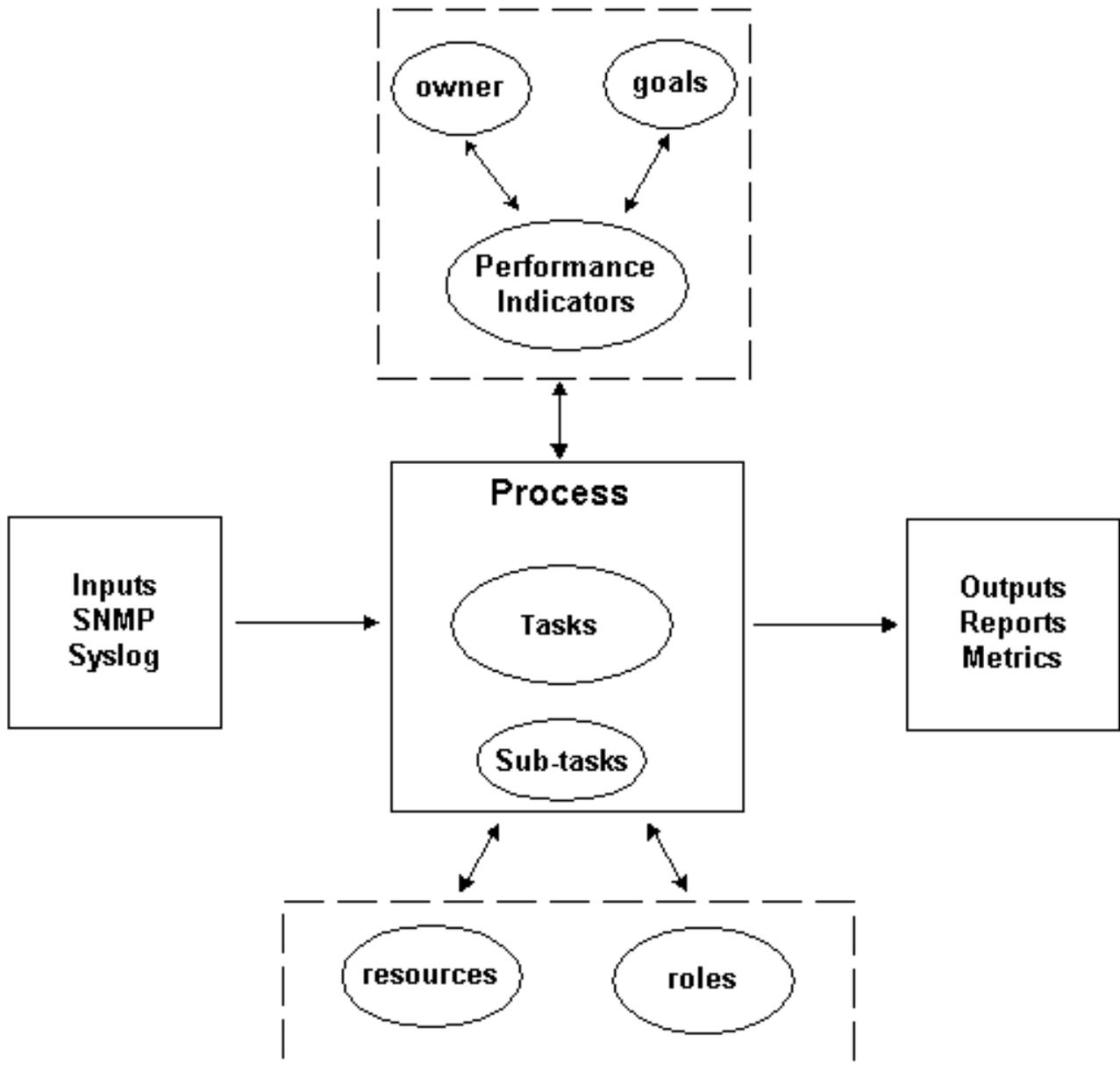
トワークのサポート	
OSPF 構成管理エリア	OSPF では、自律システムをエリアに区分することができます。これにより、さらに高いレベルのルーティング保護が行え、エリア内のルーティングはエリア外のすべての情報から保護されるようになります。また、自律システムをエリアに分割することで、CPU サイクルの面でダイクストラ処理の負担が軽減されます。
仮想リンク	OSPF では、仮想リンクの設定を可能にすることで、自律システムでのエリアのレイアウトに関するトポロジ制限を解消しています。
ルーティングプロトコル交換の認証	OSPF ルータがルーティング プロトコルのパケットを受信するたびに、オプションでそのパケットの処理前に認証を行うことができます。
柔軟に富んだルーティングメトリック	OSPF では、メトリックは発信ルータのインターフェイスに割り当てられています。このパスのコストは、そのパスを構成しているインターフェイスの合計になります。デフォルトでは、ルーティングメトリックはその回線の帯域幅から導かれます。ルーティングメトリックはネットワーク管理者によって割り当てられ、ネットワークの特性を表す遅延、帯域幅、およびコストなどと組み合わせて表現されます。
等コストマルチパス	1 つの宛先に対して最適なコストの経路が複数ある場合、OSPF ではこれらを検出し、その宛先へのトラフィックの負荷分散に使用します。
可変長サブネットのサポート	ネットワーク マスクとそれぞれのアドバタイズされた宛先を搬送することにより、可変長のサブネット マスクをサポートしています。
スタブエリアのサポート	メモリが不足しているルータをサポートするために、エリアをスタブとして設定できます。外部 LSA は、このスタブ エリアへはフラッドされません。スタブ エリアにある外部宛先へのルーティングは、デフォルトのみをベースとしています。

## プロセス定義

プロセス定義とは、特定の目的を達成するためにエージェントによって実行される一連の動作、活動、および変更を意味します。

プロセス制御とは、プロセスを効果的かつ効率的に実行することを目的とした計画および調整の過程を意味します。

図で表すと、次のようになります。



プロセスの出力は、組織によって定義された運用基準と、ビジネス上の目的に準拠している必要があります。一連の基準に準拠しているプロセスは、反復、測定、および管理が可能であり、ビジネス上の目的の達成に貢献するため、効果的であると考えられます。また、最小限の労力で活動を実行できるプロセスは、効率的であると考えられます。

## プロセス オーナー

プロセスは、さまざまな組織的境界をまたがります。したがって、プロセス定義に責任を負うプロセス所有者は 1 人だけにする必要があります。所有者は、そのプロセスが効果的および効率的であるかどうかを判断し、レポートする際を中心となります。そのプロセスが効果的または効率的ではない場合、プロセス所有者はそのプロセスの修正を余儀なくされます。プロセスを修正する際には、変更管理とチェックの手順が原則となります。

## プロセスの目的

プロセスの目標は、プロセス定義の方向付けと範囲を設定するために定められます。また、目標はプロセスの有効性を測定するためのメトリックを定義するためにも使用されます。

このプロセスの目標は、OSPF 実装の展開した構成を意図した設計に照らし合わせて検証するためのフレームワークを提供し、さらに OSPF の展開を定期的に監査して、意図した設計と長期にわたって整合性が保たれるようにするためのメカニズムを提供することです。

## プロセス パフォーマンス インジケータ

プロセス性能インジケータは、プロセス定義の有効性を測定するために使用されます。パフォーマンス インジケータは、測定および定量化が可能な基準である必要があります。次に一覧されている性能インジケータは、数値化するものか、時間で測定するものかのいずれかです。OSPF 構成管理の性能インジケータは、次のように定義されています。

- プロセス全体を一巡するために必要な時間の長さ。
- OSPF に関係する問題を、ユーザに影響を与える前に予防的に検出するために必要な実行頻度。
- プロセスの実行に関連するネットワークの負荷。
- プロセスによって推奨される修正処理の回数。
- プロセスの結果として実施された修正操作の数。
- 修正処理を実行するために必要な時間の長さ。
- 修正処理を実行するために必要な時間の長さ。
- 修正操作の未処理件数
- OSPF に関する問題に起因するダウンタイム。
- シード ファイル内で追加、削除、または修正された項目の数。これは正確性と安定性を示します。

## プロセスの入力

プロセスの入力は、プロセスの基準と前提条件を定義するために使用されます。プロセス入力の識別により、外部の依存関係に関する情報が何度も提供されます。OSPF 構成管理に関連する入力の一覧は次のとおりです。

- OSPF 設計文書
- SNMP ポーリングによって収集された OSPF MIB データ
- syslog 情報

## プロセスの出力

プロセスの出力は、次のように定義されます。

- この文書の「データの表示」セクションで定義された OSPF 構成レポート
- 修正処理を実行するための OSPF 構成の推奨

## タスク定義

以降のセクションでは、OSPF 構成管理に関連する初期化タスクと反復タスクを定義します。

### 初期化タスク

初期化タスクは、プロセスを実装するときに 1 回実行されます。プロセスが反復される際には実行しないようにします。

### 必須タスクの確認

必須タスクの確認中に、いずれかのタスクが実施されていないか、あるいはこの手順の要求に応えるだけの十分な情報が提供されていない場合は、この事実をプロセス所有者が文書化し、管理者に提出する必要があります次の表では、必須の初期化タスクを概説しています。

必須タスク	説明
タスクの目的と入力	<ol style="list-style-type: none"> <li>1. OSPF 設計文書が存在し、そのネットワーク設計文書の中に次の情報が明確に記されていることを確認します。ストレージ定義：名前、アドレス範囲とタイプエリア境界ルータおよび自律システム境界ルータ ( ABR/ASBR ) の識別 DR/BDR の識別エリアに割り当てられるインターネットレジストリ ( IR ) ノードとインターフェイス</li> <li>2. SNMP 標準構成のテンプレートを使用して、ネットワークに SNMP が構成されていることを確認します。注：これは後でシードファイルを作成するための入力として使用されます。</li> <li>3. Syslog 標準構成のテンプレートを使用して、ネットワークに Syslog が展開されていることを確認します。</li> </ol>
タスクの出力	<p>タスク出力は、必須タスクの条件に関するステータスレポートです。サポートされているタスクのいずれかが効率的でないと思われる場合は、プロセス所有者が要求を発行し、サポートされているプロセスが更新されるようにします。サポートされているプロセスを更新できない場合は、このプロセスの影響について評価を行います。</p>
タスク	ネットワーク エンジニアのスキル セット

の 役 割	
-------------	--

## シード ファイルの作成

OSPF 構成管理プロセスでは、シード ファイルを作成して、ネットワーク調査機能の必要性をなくすことが必要です。シード ファイルには、OSPF プロセスによって管理されるルータのセットが記録され、組織の変更管理プロセスとの調整の中心として使用されます。たとえば、ネットワークに新しいノードが組み込まれる場合、これらは OSPF シード ファイルに追加される必要があります。セキュリティ上の理由で SNMP コミュニティ名に変更を加えた場合は、それらの変更をシード ファイルに反映する必要があります。次の表では、シード ファイルの作成手順を概説しています。

プロセス	説明
タスクの目的	OSPF 構成管理ソフトウェアを初期化するために使用されるシード ファイルを作成します。シード ファイルの形式は、OSPF 構成管理プロセスの実装に使用されるリソースによって異なります。カスタム スクリプトが作成される場合、シード ファイルの形式は、ソフトウェアの設計によって定義されます。ネットワーク管理システム (NMS) が使用されている場合、シード ファイルの形式は、NMS の文書によって定義されています。
タスクの入力	<ol style="list-style-type: none"> <li>1. シード ファイルのフォーマットします。</li> <li>2. OSPF 設計文書を参照して、次のデータを割り出します。全ノードの IP アドレス SNMP コミュニティ スtring Telnet および CLI ログインのアカウントとパスワード</li> <li>3. ネットワーク変更管理プロセスをスケジュールし、名前を問い合わせます。</li> </ol>
タスクの出力	OSPF 構成管理プロセスのためのシード ファイル。
タスクのリソース	<ul style="list-style-type: none"> <li>• 商用 NMS システム</li> <li>• カスタム開発されたソフトウェア システム</li> <li>• 手作業による処理：各ネットワーク要素にログインし、コマンドラインを発行して、その出力を記録します。</li> </ul>
タスクの役割	<ul style="list-style-type: none"> <li>• NMS：ネットワークエンジニア、NMS 管理者、および NMS スクリプト スキル セット。</li> <li>• カスタム スクリプト：ネットワーク エンジニア および NMS スクリプト スキル セット。</li> <li>• 手動プロセス：ネットワーク エンジニア。</li> </ul>

## 反復タスク

反復タスクは、プロセスの反復ごとに実行され、性能インジケータを向上させるためにその頻度が判別され、修正されます。

## シード ファイルを管理する

シード ファイルは、OSPF 構成管理プロセスを効果的に実装するために重要なものです。したがって、シード ファイルの現在の状態を積極的に管理する必要があります。シード ファイルの内容に影響を及ぼすようなネットワークの変更は、OSPF 構成管理プロセスの所有者が追跡する必要があります。

プロセス	説明
タスクの目的	<ol style="list-style-type: none"><li>1. ネットワークの移動、追加、変更、およびネットワーク構成の変更を管理する、組織機能による追跡や相互対話を通じて、現在のシード ファイルを維持します。</li><li>2. シード ファイルに対するバージョン制御とバックアップ制御を行います。</li></ol>
タスクの入力	<ol style="list-style-type: none"><li>1. 移動、追加、変更など、シード ファイルの内容に影響を及ぼすものに関する変更管理からの情報。</li><li>2. シード ファイルの内容に影響を及ぼすエンジニアリングおよび設計からの情報。</li></ol>
タスクの出力	<ol style="list-style-type: none"><li>1. 現在のシード ファイルに関する週単位のレポート。</li><li>2. シード ファイルのバックアップの場所と復元方法に関する定義と文書。</li></ol>
タスクのリソース	<ul style="list-style-type: none"><li>• 商用 NMS システム</li><li>• カスタム開発されたソフトウェア システム</li><li>• 手作業による処理：各ネットワーク要素にログインし、コマンドラインを発行して、その出力を記録します。</li></ul>
タスクの役割	<ul style="list-style-type: none"><li>• NMS：ネットワークエンジニア、NMS管理者、およびNMSスクリプトスキルセット。</li><li>• カスタム スクリプト：ネットワーク エンジニアおよび NMS スクリプト スキル セット。</li><li>• 手動プロセス：ネットワーク エンジニア。</li></ul>

## OSPF スキャンの実行

OSPF スキャンの実行には、次の 2 つの手順が使用されます。

1. データ収集。
2. データの解析。

プロセスの使用方法によっては、これらの 2 つの手順の頻度が変わります。たとえば、このプロセスをインストールの変更の確認に使用することができます。この場合、変更の前後にデータ収集が実行され、変更後にデータの解析が実行されて、変更が正しく行われたかどうか判断されます。

このプロセスが OSPF 構成管理の設計の記録を確認するために使用される場合は、データ収集と解析の頻度は、ネットワークで変更が行われる頻度によって変化します。たとえば、ネットワーク上で行われる変更の数が著しく多い場合、設計の検証は 1 週間に 1 度行います。ネットワーク上での変更の数が非常に少ない場合は、設計の検証は 1 か月に 1 度行う程度です。

## OSPF レポートの検討

OSPF 構成管理レポートの形式は、OSPF 構成管理プロセスの実装に使用されるリソースによって異なります。次の表では、カスタム開発されたレポートの推奨される形式について説明しています。

レ ポ ー ト	書式
タ ス ク の 入 力	OSPF 構成管理レポートについては、この文書の「データの表示」セクションを参照してください。
タ ス ク の 出 力	スキャンレコードと論理設計レコードの間で問題が見つかった場合は、どちらの項目が正しく、どちらの項目が誤っているかについて、判断を下す必要があります。誤った項目は修正する必要があります。これには、設計レコードの変更やネットワークの変更順序が関係している場合があります。
タ ス ク の リ ソ ー ス	<ul style="list-style-type: none"> <li>• 商用 NMS システム</li> <li>• カスタム開発されたソフトウェアシステム</li> <li>• 手作業：各ネットワーク要素にログインし、コマンドラインを発行して、その出力を記録します。</li> </ul>
タ ス ク の 役 割	<ul style="list-style-type: none"> <li>• NMS：ネットワークエンジニア、NMS 管理者、および NMS スクリプトスキルセット。</li> <li>• カスタム スクリプト：ネットワーク エンジニア および NMS スクリプト スキル セット。</li> <li>• 手動プロセス：ネットワーク エンジニア。</li> </ul>

## データの識別

### 一般的なデータの特徴

次の表では、OSPF 構成管理に適用されるデータについて説明します。

Data	説明
------	----

OSPF エリア	<p>ルータが接続されたエリアについて説明する情報。次のものが含まれます。</p> <ul style="list-style-type: none"> <li>• エリア ID</li> <li>• エリア認証</li> <li>• SPF の実行</li> <li>• エリア内の ABR の数</li> <li>• エリア内の ASBR の数</li> <li>• エリア LSA の数：エリア内のルータ全体の整合性</li> <li>• エリア LSA のチェックサム：エリア内のルータ全体の整合性</li> <li>• エリアごとのアドレッシング エラーによるパケット廃棄の頻度</li> <li>• エリアごとのルーティング処理によるプロトコル パケットの廃棄の頻度</li> <li>• エリアごとにルートが見つからない状態によるルーテッド パケットの廃棄の頻度</li> </ul>
OSPF インターフェイス	<p>OSPF の観点から見たインターフェイスの説明。次の項目が含まれます。</p> <ul style="list-style-type: none"> <li>• IP アドレス</li> <li>• エリア ID</li> <li>• 管理ステータス</li> <li>• このインターフェイスに割り当てられた OSPF メトリック</li> <li>• このインターフェイスに割り当てられた OSPF タイマー</li> <li>• OSPF の状態</li> </ul>
OSPF 近隣ルータの状態	<p>OSPF 近隣ルータの説明。</p> <ul style="list-style-type: none"> <li>• 近接ルータのルータ ID</li> <li>• ネイバー ステート</li> <li>• 近隣ルータでのイベント：ネイバー関係により状態が変化したり、エラーが発生した回数。</li> <li>• 近隣ルータの再送信キュー：再送信キューの現在の長さ。</li> </ul>

## SNMP データの識別

シスコは現在、[RFC 1253 OSPFバージョン2 MIB](#) ( RFC 1253 OSPFバージョン2 MIB ) をサポートしています。RFC 1253には、OSPF に対する SNMP トラップの定義は含まれていません。OSPF MIBの最新バージョンは[RFC 1850 OSPF Version 2](#)。SNMPトラップは、RFC 1850でOSPFに定義されています。RFC 1850は、シスコのOSPF MIBの実装ではサポートされません。

詳細は、この文書の「SNMP ポーリング データ」のセクションを参照してください。

プラットフォームおよびコードバージョンでサポートされている MIB に関する詳細なリストについては、『[Cisco ネットワーク管理用ソフトウェア](#)』のページを参照してください。

## [RMON データの識別](#)

この手順に必要な RMON 特有のデータはありません。

## [Syslog データの識別](#)

一般的には、Syslog によって、異なるテクノロジーに対するサービス固有のメッセージが生成されます。syslog 情報は障害や性能の管理に適していますが、ここで提供される情報は参照用です。シスコのデバイスによって生成される OSPF Syslog 情報の詳細は、「OSPF エラー メッセージ」を参照してください。

ファシリティによるシステム メッセージの完全なリストについては、「メッセージと回復の手順」を参照してください。

## [Cisco IOS CLI データの識別](#)

このバージョンの OSPF 構成管理手順では、CLI データは必要ありません。

## [データ収集](#)

### [SNMP データ収集](#)

次の表では、SNMP データの収集に関する各種のコンポーネントについて説明します。

SNMP データのコンポーネント	定義
一般的な SNMP の設定	SNMP 設定のベスト プラクティスに関する一般的な情報については、『 <a href="#">SNMP の設定</a> 』を参照してください。
サービスに固有の SNMP 構成	この手順に必要なサービス固有の SNMP 構成はありません。
SNMP MIB の要件	前述の「 <a href="#">データの識別</a> 」セクションを参照してください。
SNMP MIB ポーリングの収集	SNMPポーリングされたデータは、 <a href="#">hp OpenView</a> などの商用システムまたはカスタム <a href="#">スクリプト</a> によって収集されます。収集アルゴリズムの詳細は、この文書の「データ収集アルゴリズムの例」のセクションを参照してください。
SNMP MIB ト	シスコのデバイスでサポートされている OSPF MIB の現在のバージョンでは、SNMP トラップ

ラップ の収集	をサポートしていません。この手順に必要な SNMP トラップはありません。
------------	--

## RMON データの収集

この手順の現在のバージョンでは、RMON 構成とデータは必要ありません。

## Syslog データの収集

一般的な syslog 構成のガイドラインは、この文書の範囲外です。詳細については、『[単一の内部ネットワークでの Cisco Secure PIX Firewall の設定とトラブルシューティング](#)』を参照してください。

OSPF 固有の要件は、次のコマンドを使用して、近隣ルータの変更を syslog メッセージによって記録するよう OSPF ルータを設定することです。

```
OSPF_ROUTER(config)# ospf log-adj-changes
```

## Cisco IOS CLI データの収集

一般的には、NE によって蓄積された未加工の情報に最も直接的にアクセスできるのは Cisco IOS CLI です。しかし、CLI によるアクセスは、この手順で定義されているグローバルな構成管理よりも、トラブルシューティングや変更管理のアクティビティにより適しています。CLI によるアクセスは、大規模なネットワークの管理には適していません。その場合は自動情報アクセスが必要になります。

このバージョンの OSPF 構成管理手順では、CLI 構成とデータは必要ありません。

## データ表示

### OSPF エリア レポート

OSPF エリア レポートの形式の例を次に示します。このレポートの形式は、商用 NMS のいずれかが使用されていればその機能によって、あるいは設計されているカスタム スクリプトの出力によって決まります。

領域	データ フィールド	直前の実行	今回の実行
エリア ID #1	[Authentication]		
	SPF の実行		
	ABR の回数		
	ASBR の回数		
	LSA の回数		
	LSA のチェックサム		
	アドレス エラー		
	ルーティングの廃棄		
	ルートが見つからない		
エリア	[Authentication]		

ID #n	SPF の実行		
	ABR の回数		
	ASBR の回数		
	LSA の回数		
	LSA のチェックサム		
	アドレスエラー		
	ルーティングの廃棄		
	ルートが見つからない		

## OSPF インターフェイス レポート

OSPF インターフェイス レポートの形式の例を次に示します。実際には、このレポートの形式は、商用 NMS のいずれかが使用されていればその機能によって、あるいは設計されているカスタム スクリプトの出力によって決まります。

領域	デバイス	インターフェイス	データ フィールド	直前の実行	今回の実行
エリア ID #1	ノード ID #1	インターフェイス ID #1	iSCSIポータルの		
			エリア ID		
			管理状態		
			OSPF の状態		
			メトリック/コスト/タイマー		
		インターフェイス ID #n	iSCSIポータルの		
			エリア ID		
			管理状態		
			OSPF の状態		
			メトリック/コスト/タイマー		
	ノード ID #n	インターフェイス ID #1	iSCSIポータルの		
			エリア ID		
			管理状態		
			OSPF の状態		
			メトリック/コスト/タイマー		
		インターフェイス ID #n	iSCSIポータルの		
			エリア ID		
			管理状態		
			OSPF の状態		
			メトリック/コスト/タイマー		
エリア	ノード ID #1	インターフェイス ID #1	iSCSIポータルの		
			エリア ID		
			管理状態		

ID #n	ノード ID #n	インターフェイス ID #n	OSPF の状態		
			メトリック/コスト/タイマー		
			iSCSIポータルの		
			エリア ID		
			管理状態		
		OSPF の状態			
		メトリック/コスト/タイマー			
		インターフェイス ID #1	iSCSIポータルの		
			エリア ID		
			管理状態		
	OSPF の状態				
	メトリック/コスト/タイマー				
	インターフェイス ID #n	iSCSIポータルの			
		エリア ID			
		管理状態			
OSPF の状態					
メトリック/コスト/タイマー					

## OSPF 隣接ルータ レポート

OSPF 近隣ルータ レポートの形式の例を次に示します。実際には、このレポートの形式は、商用 NMS のいずれかが使用されていればその機能によって、あるいは設計されているカスタム スクリプトの出力によって決まります。

領域	デバイス	隣接ルータ	データ フィールド	直前の実行	今回の実行
エリア ID #1	ノード ID #1	近隣ルータ ID #1	ルータ ID		
			ルータ IP アドレス		
			都道府県		
			イベント		
			再送信キュー		
		近隣ルータ ID #n	ルータ ID		
			ルータ IP アドレス		
			都道府県		
			イベント		
			再送信キュー		
	ノード ID #n	近隣ルータ ID #1	ルータ ID		
			ルータ IP アドレス		

エリア ID #n			都道府県			
			イベント			
			再送信キュー			
		近隣ルータ ID #n	ルータ ID			
			ルータIPアドレス			
			都道府県			
	ノード ID #n	ノード ID #1	近隣ルータ ID #1	ルータ ID		
				ルータIPアドレス		
				都道府県		
			イベント			
			再送信キュー			
			近隣ルータ ID #n	ルータ ID		
		ルータIPアドレス				
		都道府県				
ノード ID #n		近隣ルータ ID #1	ルータ ID			
			ルータIPアドレス			
			都道府県			
		イベント				
	再送信キュー					
	近隣ルータ ID #n	ルータ ID				
ルータIPアドレス						
都道府県						
			イベント			
			再送信キュー			

## 商用および一般的なインターネット監視ツール

商用ツールの目的は、syslog 情報の収集と処理を支援することと、一般的な SNMP MIB 変数のポーリングを収集することです。

この手順で定義した OSPF 構成管理をサポートする商用または公的なインターネット監視ツールは知られていません。したがって、ローカルなカスタム スクリプトや手順が必要になります。

## SNMP ポーリング データ

## ルート テーブル RFC 1213

Object Name	オブジェクトの説明
ipRouteDest	このルートの宛先 IP アドレス。0.0.0.0 の値を持つエントリは、デフォルト ルートと見なされます。Multiple routes to a single destination can appear in the table, but access to such multiple entries is dependent on the table-access mechanisms defined by the network management protocol in use. ::= { ipRouteEntry 1 } オブジェクト識別子 = 1.3.6.1.2.1.4.21.1.1
ipRouteMask	ipRouteDest フィールドの値と比較される前に、宛先アドレスに対して論理的にマスクがけを行うことを示します。任意のサブネット マスクをサポートしていないシステムの場合、エージェントにより、ipRouteDest フィールドがクラス A、B、C のいずれのネットワークに属するかによって、次のマスク ネットワークの中から ipRouteMask の値が構築されます。 <ul style="list-style-type: none"> <li>• クラス A = 255.0.0.0</li> <li>• クラス B = 255.255.0.0</li> <li>• クラス C = 255.255.255.0</li> </ul> ipRouteDest の値が 0.0.0.0、すなわちデフォルト ルートの場合、マスクの値も 0.0.0.0 になります。 <b>注：すべてのIPルーティングサブシステムは、このメカニズムを暗黙的に使用します。</b> ::= { ipRouteEntry 11 } object identifier = 1.3.6.1.2.1.4.21.1.11
ipRouteNextHop	このルートのネクスト ホップの IP アドレス。ブロードキャスト メディアによって実現されているインターフェイスへのルート境界の場合は、このフィールドの値はそのインターフェイスのエージェントの IP アドレスになります。 ::= { ipRouteEntry 7 } object identifier = 1.3.6.1.2.1.4.21.1.7
ipRouteIfIndex	このルートのネクスト ホップが通過するローカル インターフェイスを一意に識別するインデックス値。このインターフェイスは、IfIndex の値で識別されるインターフェイスと同一です。 ::= { ipRouteEntry 2 } オブジェクト識別子 = 1.3.6.1.2.1.4.21.1.2

## RFC 1213 の各種オブジェクト

Object Name	オブジェクトの説明
ipAdEntIf	このエントリを受け入れるインターフェイスを一意に識別するインデックス値。このインターフェ

Index	イスは、IfIndex の値で識別されるインターフェイスと同一です。 ::= { ipAddrEntry 2 } object identifier = 1.3.6.1.2.1.4.20.1.2
ipInAddrErrors	IP ヘッダー内の IP アドレスがエントリに対して誤った宛先を指していたために廃棄された入力データグラムの数。この数には、無効なアドレス ( 0.0.0.0 ) およびサポートされていないクラスアドレス ( クラス E ) も含まれます。 IP ゲートウェイでなく、データグラムを転送しないエンティティの場合、このカウンタには宛先アドレスがローカルアドレスでないために廃棄されたデータグラムが含まれます。 ip {5}オブジェクトID = 1.3.6.1.2.1.4.5
ipRoutingDiscards	廃棄された有効なルーティング エントリの数。このようなエントリを廃棄する理由のひとつとしては、バッファスペースを他のルーティング エントリに対して開放したことが考えられます。 ip {23}オブジェクトID = 1.3.6.1.2.1.4.23
ipOutputNoRoutes	宛先に転送するためのルートが見つからなかったために廃棄された IP データグラムの数。 ip {12}オブジェクトID = 1.3.6.1.2.1.4.12

## RFC 1253 OSPFエリアテーブル

Object Name	オブジェクトの説明
ospfAreaID	エリアを一意に識別する 32 ビットの整数。エリア ID 0.0.0.0 は、OSPF バックボーンのために使用されます。 ::= { ospfAreaEntry 1 }オブジェクト識別子 = 1.3.6.1.2.1.14.2.1.1
ospfAuthType	このエリアに対して指定された認証タイプ。その他の認証タイプは、エリアごとにローカルに割り当てることができます。デフォルト値は0です。 ::= { ospfAreaEntry 2 } object identifier = 1.3.6.1.2.1.14.2.1.2
OspfSpfRuns	エリア内のルート テーブルが、このエリアのリンクステート データベースを使用して計算された回数。オブジェクト識別子 = 1.3.6.1.2.1.14.2.1.4
ospfAreaBdrRtrCount	このエリア内で到達可能な ABR の回数。この値は最初はデフォルト値の 0 であり、各 SPF パスごとに計算されます。 ::= { ospfAreaEntry 5 } object identifier = 1.3.6.1.2.1.14.2.1.5
ospfASBdrRtrCount	このエリア内で到達可能な ABSR の回数。この値は最初はデフォルト値の 0 であり、各 SPF パスごとに計算されます。 ::= { ospfAreaEntry 6 } object identifier = 1.3.6.1.2.1.14.2.1.6
ospfAreaLSACount	エリアのリンクステート データベース内の LSA の合計数。外部 LSA は含まれません。デフォルト値は0です。 ::= { ospfAreaEntry 7 } object identifier = 1.3.6.1.2.1.14.2.1.7

ospfAreaLSACksumSum	<p>エリアのリンクステート データベースに含まれる LSA の LS チェックサム の合計を表す 32 ビット符号なしの値。この合計値には外部 (LS タイプ 5) の LSA は含まれません。この合計値は、ルータのリンクステート データベースに変更があったかどうかを判断するためと、2 台のルータのリンクステート データベースを比較するために使用されます。デフォルト値は 0 です。</p> <p>::= { ospfAreaEntry 8 } object identifier = 1.3.6.1.2.1.14.2.1.8</p>
---------------------	---

### RFC 1253 OSPF インターフェイス テーブル

Object Name	オブジェクトの説明
OspfIfaceAddress	OSPF インターフェイスの IP アドレス。オブジェクト識別子 = 1.3.6.1.2.1.14.7.1.1
OspfIfaceEvents	OSPF インターフェイスが自身の状態を変更した回数、あるいはエラーが発生した回数。オブジェクト識別子 = 1.3.6.1.2.1.14.7.1.15
OspfIfaceState	OSPF インターフェイスの状態。オブジェクト識別子 = 1.3.6.1.2.1.14.7.1.12

### RFC 1253 OSPF 近隣ルータ テーブル

Object Name	オブジェクトの説明
OspfNeighborAddress	この隣接ルータの IP アドレス。 ::= { ospfNeighborEntry 1 } オブジェクト ID = 1.3.6.1.2.1.14.10.1.1
ospfNeighborAddressLessIndex	IP アドレスを持たないインデックスで、インターネット標準の MIB の IfIndex に対応する値。列を作成するときに、そのインスタンスから得ることができます。 ::= { ospfNeighborEntry 2 } object identifier = 1.3.6.1.2.1.14.10.1.2
ospfNeighborRtrId	IpAddress として表現される 32 ビットの整数で、自律システムでの近隣ルータを一意に識別します。デフォルト値は 0.0.0.0 です。 ::= { ospfNeighborEntry 3 } object identifier = 1.3.6.1.2.1.14.10.1.3
ospfNeighborState	<p>近隣ルータとの関係の状態。状態には次のものがあります。</p> <ul style="list-style-type: none"> <li>• down (1)</li> <li>• attempt (2)</li> <li>• init (3)</li> <li>• twoWay (4)</li> <li>• exchangeStart (5)</li> <li>• exchange (6)</li> <li>• loading (7)</li> <li>• full (8)</li> </ul> <p>::= { ospfNeighborEntry 6 } object identifier =</p>

	1.3.6.1.2.1.14.10.1.6
ospfNbrEvents	近隣ルータの関係により状態が変化したり、エラーが発生した回数。デフォルト値は0です。 ::= { ospfNbrEntry 7 } object identifier = 1.3.6.1.2.1.14.10.1.7
ospfNbrLSRetransQLen	再送信キューの現在の長さ。デフォルト値は0です。 ::= { ospfNbrEntry 8 } object identifier = 1.3.6.1.2.1.14.10.1.8

## データ収集アルゴリズムの例

このドキュメントの調査で、プロトタイプとなる C プログラムを開発しました。oscanと呼ばれるプログラムは、Microsoft Developer Studio 97とVisual C++バージョン5.0を使用して作成されました。SNMP関数アプリケーションプログラミングインターフェイス(API)を提供する2つの特定のライブラリがあります。そのライブラリは、snmpapi.lib と mgmtapi.lib です。

この Microsoft API によって提供される関数は 3 つの主要なカテゴリに分類され、次の表に一覧されています。

エージェント関数	マネージャ関数	ユーティリティ関数
SnmpExtensionInit	SnmpMgrClose	SnmpUtilMemAlloc
SnmpExtensionInitEx	SnmpMgrGetTrap	SnmpUtilMemFree
SnmpExtensionQuery	SnmpMgrOidToStr	SnmpUtilMemReAlloc
SnmpExtensionTrap	SnmpMgrOpen	SnmpUtilOidAppend
	SnmpMgrRequest	SnmpUtilOidCmp
	SnmpMgrStrToOid	SnmpUtilOidCpy
	SnmpMgrTrapListen	SnmpUtilOidFree
		SnmpUtilOidNCmp
		SnmpUtilPrintAsnAny
		SnmpUtilVarBindCpy
		SnmpUtilVarBindListCpy
		SnmpUtilVarBindFree
		SnmpUtilVarBindListFree

oscan プロトタイプ コードでは、次に一覧されている一連の関数に、Microsoft API をカプセル化しています。

- snmpWalkStrOid
- snmpWalkAsnOid
- snmpWalkVarBind
- snmpWalkVarBindList

これらの関数は、OSPF 構成データを維持するために使用される各種の SNMP MIB テーブルにアクセスできるようにする一般的な API を提供しています。アクセスされるテーブルの object identifier (OID; オブジェクト識別子) は、テーブル固有のコールバック関数と共に oscan の API に渡されます。コールバック関数には、テーブルから返されたデータを処理するための情報が含まれます。

## メインルーチン

最初の作業は、oscan プログラムのターゲットとなるノードの一覧を構築することです。「デバイス検出」の問題を回避するには、スキャンされるノードを識別するためのシードファイルが必要です。シードファイルは、IP アドレスや SNMP の読み取り専用コミュニティストリングのような情報を提供します。

oscan プログラムでは、ルータによって収集された SNMP 情報を保存するための、いくつかの内部データ構造を維持する必要があります。一般的には、収集されたそれぞれの SNMP MIB テーブル用の内部データ構造があります。

```
Main
load node array based on information in the seed file.
while more entries in the node array
start SNMP session for this node
collect IP route table for this node
collect OSPF area table for this node
collect OSPF Neighbor table for this node
collect sysName for this node
collect OSPF Interface table for this node
end SNMP session for this node
end while
```

## IP ルート テーブル

この操作はルータの CPU に負荷をかけやすいため、SNMP によって IP ルート テーブルにアクセスする際は十分に注意してください。また、この理由から、oscan プログラムではユーザによる設定が可能な遅延パラメータを使用しています。このパラメータを使用することによって、各 SNMP 要求間での遅延が設定できます。大規模な環境の場合は、このために情報の収集にかかる総時間が非常に長くなる場合があります。

ルート テーブルには、oscan が関係する 4 つの情報があります。

- ipRouteDest
- ipRouteMask
- ipRouteNextHop
- ipRoutelfIndex

ルート テーブルは、ipRouteDest によって示されます。したがって、SNMP get-request によって返される各オブジェクトには、OID に付加された ipRouteDest が含まれます。

ipRoutelfIndex オブジェクトは、IP アドレス テーブル ( ipAddrTable ) を示す整数です。ipAddrTable は、ipAdEntAddr オブジェクト ( そのインターフェイスの IP アドレス ) を使用して示されます。インターフェイスの IP アドレスを取得するには、4 段階の処理が必要です。

1. ルーティング テーブルから ipRoutelfIndex を収集する。
2. ipRoutelfIndex を使用して ipAddrTable にアクセスし、パターン マッチングを行う。
3. パターンが見つかったら、OID をストリングに変換し、そのインターフェイスの IP アドレスとなる直前のドットで分割された 10 進数のフィールドを収集する。
4. インターフェイスの IP アドレスを IP ルート テーブルに戻す。

IP ルート テーブルにアクセスするための一般的なアルゴリズムを次に示します。このとき、ipRoutelfIndex の整数値だけが保存されます。この処理の後の方で、インターフェイス情報を収集するとき、ipAddrTable にアクセスが行われ、残りの情報が収集されて、内部 IP ルート テーブルに配置されます。

```
OID List =
ipRouteDestOID,
ipRouteMaskOID,
ipRouteNextHopOID,
ipRouteIfIndexOID;
```

```
For each object returned by SNMP route table walk
Sleep // user configurable polling delay.
check varbind oid against OID list
if OID is ipRouteDestOID
add new entry in the internal route table array
if OID is one of the others
search internal route array for matching index value
store information in array
```

収集された情報は、次のように今までに馴染みのあるルータの CLI からの出力に似た表で表現されます。

```
ROUTE TABLE
*****
Destination      Mask                GW                  Interface
10.10.10.4        255.255.255.252    10.10.10.5         10.10.10.5
10.10.10.16       255.255.255.252    10.10.10.6         10.10.10.5
10.10.10.24       255.255.255.252    10.10.10.25        10.10.10.25
10.10.10.28       255.255.255.252    10.10.11.2         10.10.11.1
10.10.10.36       255.255.255.252    10.10.10.6         10.10.10.5
10.10.11.0        255.255.255.0      10.10.11.1         10.10.11.1
10.10.13.0        255.255.255.0      10.10.11.2         10.10.11.1
```

## OSPF エリア テーブル

OSPF エリア テーブルからの情報の収集は、OSPF エリア テーブル ( ospfAreaTable ) のスキャンと、返されたデータの処理によって行われます。ospfAreaTable を示すものは ospfAreald です。ospfAreald は、IP アドレスと同様の、ドットで区分された 10 進数の形式で保存されます。したがって、ipRouteTable と ipRouteIfIndex を処理および検索するためのサブルーチンと同じサブルーチンがここで再利用できます。

このセクションの OSPF エリア テーブルには現在含まれていないデータ アイテムがいくつかあります。たとえば、ipInAddrErrors、IpRoutingDiscards、および ipOutNoRoute オブジェクトは、MIB-2 の定義に入っていますが、OSPF エリアとは関連がありません。これらのオブジェクトは、ルータと関連するものです。したがって、これらのカウンターは、エリアの各ノードの値をエリアカウンターに追加することで、エリアのメトリックとして使用されます。たとえば、OSPF エリア レポートでは、ルートが見つからないために廃棄されたパケットの数は、実際には、そのエリア内の全ルータで廃棄されたパケットの合計数になります。これはエリア内のルーティングの状態の概観を提供する高度なレベルのメトリックです。

```
OID List =
ipInAddrErrorsOID,
ipRoutingDiscardsOID,
ipOutNoRouteOID,
areaIdOID,
authTypeOID,
spfRunsOID,
abrCountOID,
asbrCountOID,
lsaCountOID,
```

```
lsaCksumSumOID;
```

```
For object returned from the SNMP walk of the Area Table
Sleep // user configurable polling delay.
check varbind oid against OID list.
if OID is ospfAreaId
add new entry in the internal route table array
if OID one of the others
search internal array for matching index value
store information in array
end of for loop
get ipInAddrErrors, ipRoutingDiscards, ipOutNoRoute
add values to overall Area counters
```

収集された情報は、次の ASCII の表で表現されます。

```
AREAS
*****
AREA = 0.0.0.0AREA = 0.0.0.2
authType = 0authType = 0
spfRuns = 38spfRuns = 18
abrCount = 2abrCount = 1
asbrCount = 0asbrCount = 0
lsaCount = 1lsaCount = 7
lsaCksumSum = 340985lsaCksumSum = 319204
ipInAddrErrors = 0 ipInAddrErrors = 0
ipRoutingDiscards = 0ipRoutingDiscards = 0
ipOutNoRoutes = 0ipOutNoRoutes = 0
```

## OSPF 隣接ルータ テーブル

近隣ルータ テーブルのインデックスには、次の 2 つの値があります。

- ospfNbrIpAddr : ospfNbrIpAddr は、ネイバーの IP アドレスです。
- ospfNbrAddressLessIndex:ospfNbrAddressLessIndex は、次の 2 つの値のいずれかです。IP アドレスが割り当てられているインターフェイスの場合、0。IP アドレスが割り当てられていないインターフェイスの場合、インターネット標準の MIB から IfIndex として変換された値。

このインデックスには 2 つの値があるため、これより前に使用した、返された OID に外部情報を付加したアルゴリズムを調整する必要があります。調整を行った後、ipRouteTable と ipRouteIfIndex を処理および検索するためのサブルーチンと同じサブルーチンがここで再利用できます。

```
OID List =
ospfNbrIpAddrOID,
ospfNbrAddressLessIndexOID,
ospfNbrRtrIdOID,
ospfNbrStateOID,
ospfNbrEventsOID,
ospfNbrLSRetransQLenOID,
```

```
For object returned from the SNMP walk of the Neighbor Table
Sleep // user configurable polling delay.
check varbind OID against OID list.
if OID matches ospfNbrIpAddr
add new entry in the internal neighbor table array
```

if OID matches one of the others  
search array for matching index value  
store information in array

収集された情報は、次の ASCII の表で表現されます。

NEIGHBORS

\*\*\*\*\*

NEIGHBOR #0NEIGHBOR #1

Nbr Ip Addr = 10.10.10.6Nbr Ip Addr = 10.10.11.2

Nbr Rtr Id = 10.10.10.17Nbr Rtr Id = 10.10.10.29

Nbr State = 8Nbr State = 8

Nbr Events = 6Nbr Events = 30

Nbr Retrans = 0Nbr Retrans = 0

## 関連情報

- [OSPF 設定ガイド](#)
- [RFC 1246 Experience with the OSPF Protocol](#)
- [RFC 1245 OSPF Protocol Analysis](#)
- [RFC 1224 Techniques for Managing Asynchronously Generated Alerts](#)
- [OSPF に関するサポート ページ](#)
- [IP ルーティングに関するサポート ページ](#)
- [テクニカルサポート - Cisco Systems](#)