



双方向フォワーディング検出 (MIB)

Bidirectional Forwarding Detection Management Information Base (draft-ietf-bfd-mib-02.txt) が実装されると、Bidirectional Forwarding Detection (BFD; 双方向フォワーディング検出) MIB のバージョン 2 機能によって、BFD 管理の Cisco IOS ソフトウェアがイネーブルにされます。BFD MIB と動作する SNMP エージェント コードによって、Cisco IOS ソフトウェアでの BFD 管理に、標準化された SNMP ベースのアプローチが使用されます。BFD MIB 機能では、CISCO-IETF-BFD-MIB が導入されます。BFD MIB は VPN に対応し、これによって、SNMP では、異なる VPN からの着信パケットが区別されます。

機能情報の確認

お使いのソフトウェア リリースが、この章で説明されている機能の一部をサポートしていないことがあります。最新の機能情報と注意事項については、ご使用のプラットフォームとソフトウェア リリースに対応したリリース ノートを参照してください。この章で説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、「[双方向フォワーディング検出 MIB の機能情報](#) (P.17) を参照してください。

プラットフォーム サポートと Cisco IOS および Catalyst OS ソフトウェア イメージ サポートに関する情報を入手するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセスします。Cisco.com のアカウントは必要ありません。

目次

- 「[双方向フォワーディング検出 MIB の制約事項](#)」 (P.2)
- 「[双方向フォワーディング検出 MIB について](#)」 (P.2)
- 「[双方向フォワーディング検出 MIB の設定方法](#)」 (P.9)
- 「[双方向フォワーディング検出 MIB の設定例](#)」 (P.11)
- 「[参考資料](#)」 (P.15)
- 「[双方向フォワーディング検出 MIB の機能情報](#)」 (P.17)

双方向フォワーディング検出 MIB の制約事項

Cisco IOS リリースの BFD MIB には、次の制約事項が適用されます。

- この MIB では、MIB オブジェクトの Read-Only (RO; 読み取り専用) 権限のみがサポートされます。ただし、`ciscoBfdSessNotificationsEnable` は例外で、SNMP `set` コマンドを介した BFD トラップをイネーブルまたはディセーブルにするために、読み取りと書き込みのアクセス権限があります。
- BFD セッション マッピング テーブル (`ciscoBfdSessMapTable`) によって、BFD セッションの複雑なインデックスが、`ciscoBfdSessionTable` で使用されるフラットな `BFDIndex` にマップされます。
- BFD では、64 ビット カウンタはサポートされません。セッション パフォーマンス テーブル (`ciscoBfdSessionPerfTable`) では、セッションごとに BFD パフォーマンス カウントが収集されます。このテーブルによって、`ciscoBfdSessionTable` は拡大されます。
- BFD MIB の VRF 対応機能は、IPv6 アドレスではサポートされません。

双方向フォワーディング検出 MIB について

ここでは、次の内容について説明します。

- 「[BFD MIB のシスコの実装機能](#)」 (P.2)
- 「[BFD MIB でサポートされる機能](#)」 (P.2)
- 「[通知生成イベント](#)」 (P.3)
- 「[双方向フォワーディング検出 MIB の利点](#)」 (P.3)
- 「[BFD MIB に関する機能とテクノロジー](#)」 (P.3)
- 「[BFD MIB でのサポート対象オブジェクト](#)」 (P.3)
- 「[VPN 対応の BFD MIB](#)」 (P.7)

BFD MIB のシスコの実装機能

BFD MIB は、`draft-ietf-bfd-mib-02.txt` という Internet Engineering Task Force (IETF) ドラフト MIB に基づいており、BFD がサポートされる機能について説明するオブジェクトが含まれます。

Cisco IOS ソフトウェア内での IETF ドラフト MIB と BFD 機能の実装とのわずかな違いは、BFD MIB と Cisco IOS ソフトウェアの内部データ構造との間で一部のマイナー変換が必要なことです。これらの変換は、ネットワークのさまざまなホストにインストールされ、動作している SNMP エージェントコードによって行われます。低い優先順位でバックグラウンドで実行されているこの SNMP エージェントコードにより、Cisco IOS ソフトウェアに対する管理インターフェイスが提供されます。

BFD MIB に定義されている SNMP オブジェクトは、すべての標準 SNMP ユーティリティによって表示されます。すべての BFD MIB オブジェクトは、IETF ドラフト MIB に基づいています。したがって、BFD MIB に関する機能および操作がサポートされる特定の Cisco SNMP アプリケーションは不要です。

BFD MIB でサポートされる機能

次の機能は、BFD MIB でサポートされます。

- BFD セッションの動作ステータスでの変更の信号を送信する通知メッセージを生成し、キューに加える機能。
- BFD MIB VPN に対応する機能。
- BFD セッションの通知メッセージをイネーブルにし、ディセーブルにし、設定する機能が提供される、既存の SNMP コマンドの拡張機能。
- 通知メッセージが送信される動作環境で、Network Management Station (NMS; ネットワーク管理ステーション) の名前または IP アドレスを指定する機能。
- 非揮発性メモリに通知設定を書き込む機能。

通知生成イベント

`session-up` キーワードおよび `session-down` キーワードを指定して、`snmp-server enable traps bfd` コマンドを使用して BFD 通知をイネーブルにする場合、Cisco IOS ソフトウェアで特定のイベントに関連する通知メッセージが生成され、ネットワークの特定の NMS に送信されます。

たとえば、BFD が設定されるときには、`bfdSessUp` 通知が NMS に送信されます。

反対に、BFD がディセーブルにされるときには、`bfdSessDown` 通知が生成され、NMS に送信されません。

双方向フォワーディング検出 MIB の利点

BFD MIB には次の利点があります。

- BFD に関する情報を取得するために、標準インターフェイス ベースの SNMP インターフェイスを提供する。
- ネットワーク管理者が評価または対処するために、指定された NMS に通知メッセージを転送する。

BFD MIB に関する機能とテクノロジー

BFD MIB 機能は、次の機能およびテクノロジーの組み合わせで使用されます。

- 標準ベースの SNMP ネットワーク管理アプリケーション
- BFD

BFD MIB でのサポート対象オブジェクト

BFD MIB には、Cisco IOS ソフトウェアで BFD 機能に対して、読み取りのみの SNMP 管理サポートを提供する、さまざまなテーブルとオブジェクトの定義が含まれています。BFD MIB は、Abstract Syntax Notation One (ASN.1; 抽象構文記法 1) に準拠しており、したがって、理想的な BFD データベースが反映されます。

いずれの SNMP ネットワーク管理アプリケーションを使用した場合でも、GET 操作を使用して BFD MIB から情報を取得し、表示できます。同様に、GETNEXT 操作を使用して、表示目的で MIB データベースに情報を移行できます。

Cisco IOS リリースでサポートされる BFD MIB テーブルおよびオブジェクトが後に続きます。重要な MIB テーブル (太字タイプでハイライト表示) については、不随のテキストで簡単に説明されています。

- セッション テーブル (**ciscoBfdSessionTable**) によって、ノードのペア間で BFD セッションが特定されます。
- セッション パフォーマンス テーブル (**ciscoBfdSessionPerfTable**) では、セッションごとに BFD パフォーマンス カウントが収集されます。このテーブルによって、**ciscoBfdSessionTable** は拡大されます。
- セッション マッピング テーブル (**ciscoBfdSessMapTable**) によって、BFD セッションの複雑なインデックスが、**ciscoBfdSessionTable** で使用されるフラットな **BFDIndex** にマップされます。

BFD 汎用変数 (スカラ)

次のパラメータは、ルータの BFD プロセスにグローバルに適用されます。

- **ciscoBfdAdminStatus** は、このルータでの BFD のグローバル管理ステータスです。イネーブルな値は、BFD プロセスが少なくとも 1 つのインターフェイスでアクティブであることを示します。ディセーブルは、いずれのインターフェイスでもイネーブルではないことを意味します。
- **ciscoBfdVersionNumber** は、BFD プロトコルの現在のデフォルトバージョン番号です。
- **ciscoBfdSessNotificationsEnable** は、True (1) に設定されている場合、**ciscoBfdSessUp** 通知と **ciscoBfdSessDown** 通知の送信がイネーブルにされます。これ以外の場合、通知は送信されません。

BFD セッション テーブル

BFD セッション テーブルによって、BFD セッション固有の情報が示され、次のエントリが含まれています。

- **ciscoBfdSessTable** によって、BFD セッションが説明されます。
- **ciscoBfdSessEntry** によって、BFD セッションが説明されます。
- **ciscoBfdSessIndex** には、このデバイスで固有の BFD セッションを表すために使用されるインデックスが含まれます。これはインデックスで、MIB ウォークにはオブジェクトとしては示されません。
- **ciscoBfdSessApplicationId** には、この BFD セッションを所有し、管理しているローカル アプリケーションを示すために使用されるインデックスが含まれます。このアプリケーション ID では、それらを管理するアプリケーションを使用してセッションを分離する便利な方法が提供されます。その値は、**show bfd client** コマンドの出力で **ClientID** に対応します。
- **ciscoBfdSessDiscriminator** によって、この BFD セッションのローカル識別子が示され、それを固有に識別するために使用されます。
- **ciscoBfdSessRemoteDiscr** によって、この BFD セッションのリモート システムによって選択されるセッション識別子が示されます。
- **ciscoBfdSessUdpPort** によって、BFD の UDP ポートが示されます。デフォルト値は、このポートでよく認知されている値です。
- **ciscoBfdSessState** によって、BFD セッションの認知ステータスが示されます。有効な値は、**adminDown** (1)、**down** (2)、**init** (3)、および **up** (4) です。
- **ciscoBfdSessRemoteHeardFlag** によって、リモート システムから BFD パケット受信のステータスが示されます。ローカル システムで、リモート システムからアクティブに BFD パケットを受信している場合、フラグは **true** (1) に設定されます。ローカル システムで BFD パケットを最近 (検出時間以内に) 受信していないか、ローカル システムが BFD セッションを分解しようとしている場

合、フラグは `false` (0) に設定されます。このオブジェクトは、セッションがバージョン 0 で実行中の場合のみ当てはまります。セッションがバージョン 1 で実行中の場合、その値によって `false` が返されます。

- `ciscoBfdSessDiag` によって、`up` (1) からその他の状態へのセッションの最後の移行に関するローカル システムでの理由を指定する、診断コードが表示されます。このオブジェクトは、通知の場合のみアクセス可能で、`ciscoBfdSessTable` の MIB ウォークでは表示されません。コードは次のとおりです。
 - `BfdInterval` : ミリ秒単位での遅延。
 - `BfdDiag` : 次の診断コード。
 - `noDiagnostic` (0)、
 - `controlDetectionTimeExpired` (1)、
 - `echoFunctionFailed` (2)、
 - `neighborSignaledSessionDown` (3)、
 - `forwardingPlaneReset` (4)、
 - `pathDown` (5)、
 - `concatenatedPathDown` (6)、
 - `administrativelyDown` (7)、
 - `reverseConcatenatedPathDown` (8)
- `ciscoBfdSessOperMode` によって、BFD セッションの現在の動作モードが示されます。サポートされる値は、次のとおりです。
 - `asyncModeWEchoFun` (1)、
 - `asynchModeWOEchoFun` (2)、
- `ciscoBfdSessDemandModeDesiredFlag` によって、デマンドモードを使用するローカル システムでの必要性が示されます。ローカル システムでデマンドモードを使用する場合は `true` (1) に設定され、使用しない場合は `false` (0) に設定されます。デマンドモードはサポートされませんので、常に、値の 0 が返されます。
- `ciscoBfdSessEchoFuncModeDesiredFlag` によって、ローカル システムでエコーモードを使用する必要があることが示されます。ローカル システムでエコーモードを使用する場合は `true` (1) に設定され、使用しない場合は `false` (0) に設定されます。
- `ciscoBfdSessControlPlanIndepFlag` によって、ローカル システムがコントロールプレーンにより割り込み中に動作できるかが示されます。ローカル システム BFD の実装が、コントロールプレーンに依存していない場合、`true` (1) に設定されます。そうではない場合、値は `false` (0) に設定されます。この値は、常に、値 0 を返します。
- `ciscoBfdSessAddrType` によって、この BFD セッションに関連付けられているインターフェイスの IP アドレスが示されます。`unknown` (0)、`ipv4` (1)、または `ipv6` (2) の値のみがサポートされます。発信インターフェイスのタイプがポイントツーポイントか、または、BFD セッションが特定のインターフェイスに関連付けられていない場合にのみ、値 `unknown` (0) を使用できます。
- `ciscoBfdSessAddr` によって、この BFD セッションに関連付けられているインターフェイスの IP アドレスが示されます。BFD セッションが特定のインターフェイスに関連付けられていない場合、値はゼロに設定されます。
- `ciscoBfdSessDesiredMinTxInterval` によって、BFD 制御パケットを送信するときにローカル システムによって使用される最小間隔が、ミリ秒単位で示されます。
- `ciscoBfdSessReqMinRxInterval` によって、ローカル システムがサポートできる受信 BFD コントロール パケット間の最小間隔が、ミリ秒単位で示されます。
- `ciscoBfdSessReqMinEchoRxInterval` によって、このシステムがサポートできる受信 BFD エコーパケット間の最小間隔が、ミリ秒単位で示されます。セッションで設定されているインターフェイスでエコーモードがディセーブルの場合、このオブジェクトは値 0 を返します。

- `ciscoBfdSessDetectMult` によって、検出時間係数が示されます。
- `ciscoBfdSessStorType` によって、このオブジェクトのストレージタイプが示されます。このエントリのストレージタイプは、常に揮発の、読み取りのみの実装です。
- `ciscoBfdSessRowStatus`。このオブジェクトは、常にアクティブの、読み取りのみの実装です。
- `ciscoBfdSessAuthPresFlag` によって、ローカルシステムで認証を使用する必要があることが示されます。ローカルシステムで認証される必要がある場合は `true` (1) に設定され、認証されない場合は `false` (0) に設定されます。認証はサポートされませんので、このオブジェクトは常に値の 0 を返します。
- `ciscoBfdSessAuthenticationType` によって、この BFD セッションで使用される認証タイプが示されます。このフィールドは、認証存在ビットが設定されている場合にのみ、有効です。このオブジェクトは、Cisco IOS の BFD では有効ではありません。

BFD セッション パフォーマンス テーブル

`ciscoBfdSessPerfTable` によって、BFD セッションのパフォーマンス カウンタが示され、`ciscoBfdSessionTable` が拡大されます。このテーブルには、次のエントリが含まれます。

- `ciscoBfdSessPerfEntry` には、BFD セッションごとに、BFD が有効なノードによって作成されたエントリが含まれます。`ciscoBfdCounterDiscontinuityTime` は、このテーブルにあるすべてのカウンタについて、不連続事象を示す場合に使用されます。
- `ciscoBfdSessPerfPktIn` によって、この BFD セッションで受信した BFD メッセージの合計数が示されます。
- `ciscoBfdSessPerfPktOut` によって、この BFD セッションで送信された BFD メッセージの合計数が示されます。
- `ciscoBfdSessUpTime` によって、セッションの最終アップ時点での、`sysUpTime` の値が示されます。このようなアップ イベントがない場合、値はゼロに設定されます。
- `ciscoBfdSessPerfLastSessDownTime` によって、ネイバーとの通信が最後に失われた時点での、`sysUpTime` の値が示されます。このようなダウン イベントがない場合、値はゼロに設定されます。
- `ciscoBfdSessPerfLastCommLostDiag` によって、ネイバーとの通信が最後に失われたときの、BFD 診断コードが示されます。このオブジェクトはサポートされません。
- `ciscoBfdSessPerfSessUpCount` によって、ルータが最後にリポートされて以降、セッションがアップの状態になった回数が示されます。
- `ciscoBfdSessPerfDiscTime` によって、1 つまたは複数のセッション カウンタで不連続事象が検出された最後の時点での、`sysUpTime` の値が示されます。同等のカウンタは、`BfdSessPerfTable` に含まれるすべての `Counter32` オブジェクトのこの BFD セッションに関連付けられている特定のインスタンスです。ローカル管理サブシステムの最後の再初期化以降に発生した不連続事象が発生しなかった場合、値はゼロです。このオブジェクトはサポートされません。
- `ciscoBfdSessPerfPktInHC` によって、この BFD セッションで受信した BFD メッセージの合計数が示されます。RFC2863 に記載されている規則に従って `ciscoBfdSessPerfPktInHC` がサポートされる場合、`ciscoBfdSessPerfPktIn` の少なくとも下位 32 ビットと等しい値である必要があります。
- `ciscoBfdSessPerfPktOutHC` によって、この BFD セッションで送信された BFD メッセージの合計数が示されます。RFC2863 に記載されている規則に従って `ciscoBfdSessPerfPktOutHC` がサポートされる場合、`ciscoBfdSessPerfPktIn` の少なくとも下位 32 ビットと等しい値である必要があります。

BFD セッション マッピング テーブル

BFD セッション マッピング テーブルによって、BFD セッションの複雑なインデックスが、`ciscoBfdSessionTable` で使用されるフラットな `BfdIndex` にマップされます。`ciscoBfdSessAddr` (OID) の値に、111 よりも多いサブ ID がある場合、このテーブルのカラム インスタンスの OID は、128 よりも多いサブ ID があり、SNMPv1、SNMPv2c、または SNMPv3 を使用した場合にはアクセスできません。BFD セッション マッピング テーブルには、次のエントリが含まれています。

- `ciscoBfdSessMapEntry` によって、このインデックスにマップされる BFD セッションが示されません。`mplsInSegmentMapLabelPtrIndex` (OID) の値に、111 よりも多いサブ ID がある場合、このテーブルのカラム インスタンスの OID は、128 よりも多いサブ ID があり、SNMPv1、SNMPv2c、または SNMPv3 を使用した場合にはアクセスできません。
- `ciscoBfdSessMapBfdIndex` によって、この行のインデックスによって参照される `BfdIndex` が示されます。本質的に、マッピングは、インデックスと `ciscoBfdSessTable` との間で提供されます。これはインデックスで、MIB ウォークにはオブジェクトとしては示されません。

マッピングの例については、「[BFD MIB の MIB ウォーク : 例](#)」(P.13) を参照してください。

通知

通知には、次のエントリが含まれます。この通知の範囲モードは、サポートされません。したがって、このセッションを表す `ciscoBfdSessTable` エントリの 1 つに、1 つの通知のみが送信されます。

- `ciscoBfdSessTable` にある 1 つまたは複数のエントリに対する `ciscoBfdSessState` オブジェクトが、他のステートから `up` (4) ステートに変わろうとしているときに、`ciscoBfdSessUp` によって通知が生成されます。`ciscoBfdSessDiag` の値は、`noDiagnostic(0)` に等しく設定されます。
- `ciscoBfdSessTable` にある 1 つまたは複数のエントリに対する `ciscoBfdSessState` オブジェクトが、他のステートから `down` (2) ステートまたは `adminDown` (1) ステートに変わろうとしているときに、`ciscoBfdSessDown` によって通知が生成されます。`ciscoBfdSessDiag` の値によって、新しいステート (つまり、`pathDown` (5) または `administrativelyDown` (7)) の理由を示す診断コードが返されます。

VPN 対応の BFD MIB

VPN 対応の BFD MIB 機能によって、BFD MIB がイネーブルにされ、VPN コンテキスト情報が取得されます。この機能によって、異なる MPLS VPN に対する異なるコンテキストへのサポートが追加されます。MIB のユーザは、指定した MPLS VPN の BFD プロセスを表示できます。VPN 対応の BFD MIB 機能では、BFD MIB の構文は変更されません。テーブル内のエントリの数とタイプが変更されません。

BFD MIB によって、一度に 1 つのコンテキストのみについて、情報を表示できます。Cisco IOS Release 12.2(33)SRE では、SNMP セキュリティ名を使用して、コンテキストを指定できます。

BFD MIB VPN 対応を作成する方法については、『[SNMP Support over VPNs—Context-Based Access](#)』の資料を参照してください。

SNMP コンテキスト

SNMP コンテキストによって、MIB データにアクセスする安全な方法が VPN ユーザに提供されます。VPN がコンテキストに関連付けられると、VPN 固有の MIB データがそのコンテキストに存在します。VPN をコンテキストに関連付けると、サービス プロバイダーが、複数 VPN でネットワークを管理できます。VPN でコンテキストを作成し、関連付けると、プロバイダーは、1 つの VPN のユーザが、同じネットワーク デバイスの他の VPN のユーザに関する情報にアクセスすることを、防ぐことができます。

VPN 対応の SNMP では、VPN 環境で動作する SNMP マネージャとエージェントのエンティティが、SNMP セキュリティ名と VPN 名との間のマッピングに一致する必要があります。このマッピングは、異なる VPN の SNMP データの異なるコンテキストを使用して作成され、これは、SNMP View-based Access Control Model MIB (SNMP-VACM-MIB) の設定を使用して行えます。セキュリティ名を使用した VPN にあるユーザが、その VPN のみのコンテキスト内にある制約オブジェクト領域にアクセスできるように、SNMP-VACM-MIB がビューを使用して設定されます。

応答メッセージが、VPN コンテキスト内のオブジェクト値を使用して返信される前に、SNMP 要求メッセージは、セキュリティとアクセス制御の次のような 3 つのフェーズを経ます。

- 1 つ目のセキュリティ フェーズは、ユーザ名の認証です。このフェーズの間、ユーザには、SNMP アクセスが許可されます。
- 2 つ目のフェーズはアクセス制御です。このフェーズの間、ユーザには、必要な SNMP コンテキストにあるグループ オブジェクトに対する、SNMP アクセスが許可されます。
- 3 つ目のフェーズでは、ユーザは、テーブル エントリの特定のインスタンスにアクセスできます。この 3 つ目のフェーズでは、完全な取得は、SNMP コンテキスト名を使用して行えます。

IP アクセス リストを設定し、SNMP コミュニティ ストリングに関連付けることができます。この機能を使用すると、ユーザは、VPN Routing and Forwarding (VRF; VPN ルーティングおよび転送) インスタンスと SNMP コミュニティ ストリングとの間の関連付けを設定できます。VRF インスタンスが SNMP コミュニティ ストリングに関連付けられると、設定された VRF から受信した場合にのみ、SNMP によって、特定のコミュニティ ストリングに着信する要求が処理されます。着信パケットに含まれるコミュニティ ストリングに、関連付けられている VRF がない場合、非 VRF インターフェイスを介して着信した場合にのみ、処理されます。

VPN 対応の BFD MIB セッション

VPN 対応の BFD MIB 機能によって、グローバル コンテキストと VPN コンテキストの両方の BFD MIB に対する SNMP クエリがサポートされます。ユーザは、すべての VRF およびコア (グローバル コンテキスト) にある BFD クエリを開始できます。クエリでは、異なる VPN からの BFD セッションが区別されます。VPN の BFD セッション情報は、VPN のコンテキストに残ります。したがって、1 つの VPN からの情報は、異なる VPN のユーザが使用することはできません。

MPLS VPN では、サービスの Provider Edge Router (PE; プロバイダー エッジ ルータ) に、いくつかの VPN およびグローバル ルーティング テーブルの VRF が含まれることがあります。同じデバイス上の異なる VPN に別の BFD プロセスを設定するには、固有の securityName、contextName、および View-based Access Control Model (VACM) ビューを使用して、各 VPN を設定します。VPN securityName は、BFD MIB に対して設定する必要があります。



(注)

特定の VPN の BFD セッション情報を確認するには、**show bfd neighbor vrf vpn-name detail** コマンドを使用します。

VPN 対応の BFD MIB 通知

VPN 対応の BFD MIB 機能によって、BFD の複数の VPN コンテキストの BFD 通知がサポートされます。BFD 通知は、グローバル コンテキストおよび異なる VPN に対して生成できます。異なる BFD コンテキストの異なる NMS ホストに、通知が送信されるよう設定できます。特定の VRF に関連付けられている BFD 数値は、その VRF に対して指定された NMS に送信されます。BFD グローバル通知は、グローバル トラップを受信するよう設定された NMS に送信されます。

VPN 対応の BFD MIB 機能の BFD コンテキスト通知をイネーブルにするには、SNMP `ciscoBfdSessNotificationEnable object` (グローバル BFD コンテキストのみ) または次の拡張グローバル コンフィギュレーション コマンドのいずれかを使用します。

グローバル コンテキストの BFD 通知をイネーブルにするには、プロバイダー エッジ (PE) ルータで次のコマンドを使用します。

```
Router(config)# snmp-server host host-address traps community
```

```
Router(config)# snmp-server enable traps bfd
```

VPN コンテキストの BFD 通知をイネーブルにするには、次のコマンドを使用します。

```
Router(config)# snmp-server host host-address vrf vrf-name version {v1|v2c|v3}
community community-string udp-port upd-port bfd
```

```
Router(config)# snmp-server enable traps bfd
```

双方向フォワーディング検出 MIB の設定方法

ここでは、次の作業について説明します。

- 「BFD MIB 通知の SNMP エージェントをイネーブルにする」(P.9) (必須)
- 「SNMP エージェントのステータスの確認」(P.10) (任意)

BFD MIB 通知の SNMP エージェントをイネーブルにする

BFD MIB の SNMP エージェントは、デフォルトでディセーブルにされています。BFD MIB 通知の SNMP エージェントをイネーブルにするには、次の手順を実行します。

手順の概要

1. `enable`
2. `show running-config | include snmp`
3. `configure terminal`
4. `snmp-server community string [view view-name] [ro | rw] [ipv6 nacl] [access-list-number]`
5. `snmp-server enable traps bfd [session-up] [session-down]`
6. `exit`
7. `write memory`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例： Router# 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	<code>show running-config include snmp</code> 例： Router# show running-config include snmp	SNMP エージェントがすでに実行中かどうか判別される実行コンフィギュレーションが表示されます。 • SNMP 情報が表示されない場合、手順 4 に進みます。SNMP 情報が表示される場合、必要に応じて、情報を変更できます。
ステップ 3	<code>configure terminal</code> 例： Router# 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 4	<code>snmp-server community string [view view-name] [ro rw] [ipv6 nacl] [access-list-number]</code> 例： Router(config)# snmp-server community comaccess ro 4	コミュニティ スtring をイネーブルにします。 • この例では、コミュニティ スtring <code>comaccess</code> を使用して <code>snmp</code> および読み取り専用アクセスがイネーブルに設定されます。
ステップ 5	<code>snmp-server enable traps bfd [session-up] [session-down]</code> 例： Router(config)# snmp-server enable traps bfd	SNMP 通知を送信するか、または SNMP ホストに通知するよう、ルータをイネーブルにします。 (注) このコマンドはオプションです。SNMP がイネーブルにされた後で、ユーザは、すべての MIB を使用して、クエリを送信できます。
ステップ 6	<code>exit</code> 例： Router(config)# exit	グローバル コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。
ステップ 7	<code>write memory</code> 例： Router# write memory	変更された設定を NVRAM に書き込み、設定を永続的に保存します。

SNMP エージェントのステータスの確認

ホスト ネットワーク デバイス上で SNMP エージェントがイネーブルにされたことを確認するには、次の手順を実行します。

手順の概要

1. `enable`
2. `show running-config | include snmp`

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： Router# 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	show running-config include snmp 例： Router# show running-config include snmp	ターゲット デバイス上で実行コンフィギュレーションが表示され、表示された SNMP 情報の出力を調べるために、使用されます。
ステップ3	show bfd neighbors detail 例： Router# show bfd neighbors detail	各ネイバーの BFD プロトコル パラメータおよびタイマーが表示されます。

例

次に、ターゲット デバイスおよびその SNMP 情報の実行コンフィギュレーションが表示されます。

```
Router# show running-config | include snmp
.
.
.
snmp-server community public rw
snmp-server community private ro
```

すべての **snmp-server** ステートメントが出力に表示され、そのデバイス上で SNMP がイネーブルにされたことを確認するために、次の形式で示されます。

双方向フォワーディング検出 MIB の設定例

ここでは、次の設定例を示します。

- 「SNMP エージェントをイネーブルにして BFD 通知をイネーブルにする：例」 (P.11)
- 「BFD セッションの表示：例」 (P.12)
- 「BFD MIB の MIB ウォーク：例」 (P.13)

SNMP エージェントをイネーブルにして BFD 通知をイネーブルにする：例

次に、ホスト ネットワーク デバイスで SNMP エージェントをイネーブルにする例を示します。

```
Router# configure terminal
Router(config)# snmp-server community private ro
```

次の例では、*comaccess* コミュニティ スtring が指定されているアクセス リスト 4 のメンバーに関連するすべての BFD MIB オブジェクトに対して、読み取りアクセスを許可する方法を示します。他の SNMP エージェントでは、いずれの BFD MIB オブジェクトにもアクセスできません。

```
Router(config)# snmp-server community comaccess ro 4
```

次の例は、BFD 関連の SNMP 通知を送信するか、SNMP ホストに通知するよう、ルータをイネーブルにする方法を示します。

```
Router(config)# snmp-server enable traps bfd
```

BFD セッションの表示 : 例

次に、**show bfd neighbors** コマンドの出力例を示します。これによって、各ネイバーの BFD セッションおよびタイマーが表示されます。

```
Router# show bfd neighbors
```

NeighAddr	LD/RD	RH/RS	State	Int
10.0.0.2	7/7	Up	Up	Et1/2.2
10.1.0.2	6/6	Up	Up	Et1/2.5
DDDD::1	1/1	Up	Up	Et1/3

```
Router# show bfd neighbors detail
```

NeighAddr	LD/RD	RH/RS	State	Int
10.0.0.2	9/8	Up	Up	Gi3/8.1

Session state is UP and using echo function with 50 ms interval.

OurAddr: 10.0.0.1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 5

Received MinRxInt: 1000000, Received Multiplier: 5

Holddown (hits): 0(0), Hello (hits): 1000(350)

Rx Count: 352, Rx Interval (ms) min/max/avg: 1/1000/874 last: 464 ms ago

Tx Count: 351, Tx Interval (ms) min/max/avg: 756/1000/876 last: 524 ms ago

Elapsed time watermarks: 0 0 (last: 0)

Registered protocols: CEF OSPF

Uptime: 00:05:07

Last packet: Version: 1 - Diagnostic: 0

State bit: Up - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 5 - Length: 24

My Discr.: 8 - Your Discr.: 9

Min tx interval: 1000000 - Min rx interval: 1000000

Min Echo interval: 50000

NeighAddr	LD/RD	RH/RS	State	Int
10.1.0.2	6/6	Up	Up	Gi3/8.2

Session state is UP and using echo function with 50 ms interval.

OurAddr: 10.1.0.1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 5

Received MinRxInt: 1000000, Received Multiplier: 5

Holddown (hits): 0(0), Hello (hits): 1000(352)

Rx Count: 352, Rx Interval (ms) min/max/avg: 1/1000/880 last: 248 ms ago

Tx Count: 354, Tx Interval (ms) min/max/avg: 1/1000/875 last: 244 ms ago

Elapsed time watermarks: 0 0 (last: 0)

Registered protocols: CEF OSPF

Uptime: 00:05:09

Last packet: Version: 1 - Diagnostic: 0

```

State bit: Up           - Demand bit: 0
Poll bit: 0            - Final bit: 0
Multiplier: 5         - Length: 24
My Discr.: 6          - Your Discr.: 6
Min tx interval: 1000000 - Min rx interval: 1000000
Min Echo interval: 50000
    
```

BFD MIB の MIB ウォーク : 例

次に、BFD MIB の MIB ウォークからの出力例を示します。

ciscoBfdSessMapTable

```

ciscoBfdSessMapBfdIndex.1.7.1.4.10.1.0.1 = 65543
ciscoBfdSessMapBfdIndex.3.1.2.16.221.221.0.0.0.0.0.0.0.0.0.0.0.2 = 196609
ciscoBfdSessMapBfdIndex.4.6.1.4.40.4.0.1 = 262150
The MapTable index includes the following information about BFD sessions and clients:
Index example: 1.7.1.4.10.1.0.1
1 - Client id
7 - Local discriminator
1 - IP address type (1 - IPv4, 2- IPv6)
4 - Length of next string (4 for IPv4 addresses or 16 for IPv6 addresses)
10.1.0.1 - IP address of the BFD session
    
```

ciscoBfdSessTable

```

ciscoBfdSessApplicationId.65543 = 1
ciscoBfdSessApplicationId.196609 = 3
ciscoBfdSessApplicationId.262150 = 4
ciscoBfdSessDiscriminator.65543 = 7
ciscoBfdSessDiscriminator.196609 = 1
ciscoBfdSessDiscriminator.262150 = 6
ciscoBfdSessRemoteDiscr.65543 = 7
ciscoBfdSessRemoteDiscr.196609 = 1
ciscoBfdSessRemoteDiscr.262150 = 6
ciscoBfdSessUdpPort.65543 = 3785
ciscoBfdSessUdpPort.196609 = 3784
ciscoBfdSessUdpPort.262150 = 3785
ciscoBfdSessState.65543 = up
ciscoBfdSessState.196609 = up
ciscoBfdSessState.262150 = up
ciscoBfdSessRemoteHeardFlag.65543 = false
ciscoBfdSessRemoteHeardFlag.196609 = false
ciscoBfdSessRemoteHeardFlag.262150 = false
ciscoBfdSessOperMode.65543 = asyncModeWEchoFun
ciscoBfdSessOperMode.196609 = asynchModeWEOchoFun
ciscoBfdSessOperMode.262150 = asyncModeWEchoFun
ciscoBfdSessDemandModeDesiredFlag.65543 = false
ciscoBfdSessDemandModeDesiredFlag.196609 = false
ciscoBfdSessDemandModeDesiredFlag.262150 = false
ciscoBfdSessEchoFuncModeDesiredFlag.65543 = true
ciscoBfdSessEchoFuncModeDesiredFlag.196609 = false
ciscoBfdSessEchoFuncModeDesiredFlag.262150 = true
ciscoBfdSessControlPlanIndepFlag.65543 = false
ciscoBfdSessControlPlanIndepFlag.196609 = false
ciscoBfdSessControlPlanIndepFlag.262150 = false
ciscoBfdSessAddrType.65543 = ipv4
ciscoBfdSessAddrType.196609 = ipv6
ciscoBfdSessAddrType.262150 = ipv4
ciscoBfdSessAddr.65543 = 28:01:00:01
ciscoBfdSessAddr.196609 = DD:DD:00:00:00:00:00:00:00:00:00:00:00:00:00:02
ciscoBfdSessAddr.262150 = 10:04:00:01
    
```

```

ciscoBfdSessDesiredMinTxInterval.65543 = 1000000
ciscoBfdSessDesiredMinTxInterval.196609 = 50000
ciscoBfdSessDesiredMinTxInterval.262150 = 1000000
ciscoBfdSessReqMinRxInterval.65543 = 1000000
ciscoBfdSessReqMinRxInterval.196609 = 50000
ciscoBfdSessReqMinRxInterval.262150 = 1000000
ciscoBfdSessReqMinEchoRxInterval.65543 = 50000
ciscoBfdSessReqMinEchoRxInterval.196609 = 0
ciscoBfdSessReqMinEchoRxInterval.262150 = 50000
ciscoBfdSessDetectMult.65543 = 5
ciscoBfdSessDetectMult.196609 = 5
ciscoBfdSessDetectMult.262150 = 5
ciscoBfdSessStorType.65543 = volatile
ciscoBfdSessStorType.196609 = volatile
ciscoBfdSessStorType.262150 = volatile
ciscoBfdSessRowStatus.65543 = active
ciscoBfdSessRowStatus.196609 = active
ciscoBfdSessRowStatus.262150 = active
ciscoBfdSessAuthPresFlag.65543 = false
ciscoBfdSessAuthPresFlag.196609 = false
ciscoBfdSessAuthPresFlag.262150 = false
ciscoBfdSessAuthenticationType.65543 = 0
ciscoBfdSessAuthenticationType.196609 = 0
ciscoBfdSessAuthenticationType.262150 = 0

```

ciscoBfdSessPerfTable

```

ciscoBfdSessPerfPktIn.65543 = 246
ciscoBfdSessPerfPktIn.196609 = 5159
ciscoBfdSessPerfPktIn.262150 = 290
ciscoBfdSessPerfPktOut.65543 = 247
ciscoBfdSessPerfPktOut.196609 = 5416
ciscoBfdSessPerfPktOut.262150 = 291
ciscoBfdSessUpTime.65543 = 43376
ciscoBfdSessUpTime.196609 = 39781
ciscoBfdSessUpTime.262150 = 39736
ciscoBfdSessPerfLastSessDownTime.65543 = 0
ciscoBfdSessPerfLastSessDownTime.196609 = 0
ciscoBfdSessPerfLastSessDownTime.262150 = 0
ciscoBfdSessPerfLastCommLostDiag.65543 = 0
ciscoBfdSessPerfLastCommLostDiag.196609 = 0
ciscoBfdSessPerfLastCommLostDiag.262150 = 0
ciscoBfdSessPerfSessUpCount.65543 = 1
ciscoBfdSessPerfSessUpCount.196609 = 1
ciscoBfdSessPerfSessUpCount.262150 = 1
ciscoBfdSessPerfDiscTime.65543 = 0
ciscoBfdSessPerfDiscTime.196609 = 0
ciscoBfdSessPerfDiscTime.262150 = 0
ciscoBfdSessPerfPktInHC.65543 = 247
ciscoBfdSessPerfPktInHC.196609 = 5179
ciscoBfdSessPerfPktInHC.262150 = 291
ciscoBfdSessPerfPktOutHC.65543 = 248
ciscoBfdSessPerfPktOutHC.196609 = 5440
ciscoBfdSessPerfPktOutHC.262150 = 292

```

参考資料

ここでは、双方向フォワーディング検出 MIB に関する参考資料について説明します。

関連資料

関連項目	参照先
BFD	<ul style="list-style-type: none"> 双方向フォワーディング検出
VPN に対する SNMP サポートの設定	<ul style="list-style-type: none"> 『SNMP Support over VPNs?Context-Based Access』

規格

規格	タイトル
draft-ietf-bfd-mib-03	『Bidirectional Forwarding Detection MIB』

MIB

MIB	MIB リンク
BFD MIB	選択したプラットフォーム、Cisco IOS リリース、および機能セットの MIB の場所を検索しダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
RFC 2026	「The Internet Standards Process」

シスコのテクニカル サポート

説明	リンク
<p>右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。</p> <p>以下を含むさまざまな作業にこの Web サイトが役立ちます。</p> <ul style="list-style-type: none"> ・テクニカル サポートを受ける ・ソフトウェアをダウンロードする ・セキュリティの脆弱性を報告する、またはシスコ製品のセキュリティ問題に対する支援を受ける ・ツールおよびリソースへアクセスする <ul style="list-style-type: none"> - Product Alert の受信登録 - Field Notice の受信登録 - Bug Toolkit を使用した既知の問題の検索 ・Networking Professionals (NetPro) コミュニティで、技術関連のディスカッションに参加する ・トレーニング リソースへアクセスする ・TAC Case Collection ツールを使用して、ハードウェアや設定、パフォーマンスに関する一般的な問題をインタラクティブに特定および解決する <p>この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。</p>	<p>http://www.cisco.com/en/US/support/index.html</p>

双方向フォワーディング検出 MIB の機能情報

表 1 は、この MIB のリリース履歴です。

一部のコマンドは、お使いの Cisco IOS ソフトウェア リリースで使用できないことがあります。特定の
コマンドのリリース情報については、コマンドリファレンスマニュアルを参照してください。

プラットフォーム サポートとソフトウェア イメージ サポートに関する情報を入手するには、Cisco
Feature Navigator を使用します。Cisco Feature Navigator を使用すると、Cisco IOS および Catalyst
OS ソフトウェア イメージがサポートする特定のソフトウェア リリース、機能セット、またはプラッ
トフォームを確認できます。Cisco Feature Navigator には、<http://www.cisco.com/go/cfn> からアクセス
します。Cisco.com のアカウントは必要ありません。



(注) 表 1 には、一連の Cisco IOS ソフトウェア リリースのうち、特定の機能が初めて導入された
Cisco IOS ソフトウェア リリースだけが記載されています。その機能は、特に断りがない限り、それ以
降の一連の Cisco IOS ソフトウェア リリースでもサポートされます。

表 1 双方向フォワーディング検出 MIB の機能情報

機能名	リリース	機能情報
双方向フォワーディング検出 MIB、バージョ ン 2	12.2(33)SRE	双方向フォワーディング検出 MIB 機能を使用すると、 CISCO-IETF-BFD-MIB として実装されている BFD 管理 用の Cisco IOS ソフトウェアで、SNMP エージェント サ ポートがイネーブルにされます。 次のコマンドが導入または変更されました。 <ul style="list-style-type: none"> • <code>snmp-server enable traps bfd</code> • <code>snmp-server host</code>

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

このマニュアルで使用している IP アドレスは、実際のアドレスを示すものではありません。マニュアル内の例、コマンド出力、および図は、説明のみを目的として使用されています。説明の中に実際のアドレスが使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

© 2009–2010 Cisco Systems, Inc.
All rights reserved.

Copyright © 2009–2011, シスコシステムズ合同会社.
All rights reserved.