

# 目次

[概要](#)

[EIGRP の動作理論](#)

[プロトコルの大きな変更点](#)

[基本的な理論](#)

[隣接ルータの検出と維持](#)

[トポロジ テーブルの作成](#)

[EIGRP メトリック](#)

[到達可能距離、報告距離、およびフィージブル サクセサ](#)

[パスにループがないかどうかを判断する](#)

[スプリット ホライズンとポイズン リバース](#)

[起動モード](#)

[トポロジ テーブルの変更](#)

[クエリー](#)

[Stuck In Active ルート](#)

[SIA ルートのトラブルシューティング](#)

[再配布](#)

[2 つの EIGRP 自律システム \( AS \) 間の再配布](#)

[2 つの異なる AS にある EIGRP と IGRP 間の再配布](#)

[同じ AS にある EIGRP と IGRP 間の再配布](#)

[その他のプロトコルとの間の再配布](#)

[インターフェイスへのスタティックルートの再配布](#)

[集約](#)

[自動集約](#)

[手動集約](#)

[外部ルートの自動集約](#)

[クエリーの処理と範囲](#)

[集約ポイントがクエリー範囲に与える影響](#)

[AS の境界がクエリー範囲に与える影響](#)

[配布リストがクエリー範囲に与える影響](#)

[パケットのペーシング](#)

[デフォルト ルーティング](#)

[ロード バランシング](#)

[メトリックの使用](#)

[再配布における管理タグの使用](#)

[EIGRP コマンド出力について](#)

[show ip eigrp traffic](#)

[show ip eigrp topology](#)

[show ip eigrp topology <network>](#)

[show ip eigrp topology \[active | pending | zero-successors\]](#)

[show ip eigrp topology all-links](#)

[関連情報](#)

## 概要

Enhanced IGRP ( EIGRP ) は、さまざまなトポロジやメディアに適した Interior Gateway Protocol ( IGP; 内部ゲートウェイプロトコル ) です。高度に設計されたネットワークでは、EIGRP は適切にスケールし、きわめて高速なコンバージェンス タイムを最低限のネットワークトラフィックで提供します。

## EIGRP の動作理論

数ある EIGRP の利点のいくつかを次に示します。

通常の動作では EIGRP のネットワーク リソース使用率は非常に低くなります。安定したネットワークでは hello パケットだけが送信されます。

変更が生じた場合、ルーティング テーブル全体ではなくルーティング テーブルの変更だけが伝搬されます。これにより、ネットワーク上でのルーティング プロトコル自体による負荷が減ります。

ネットワーク トポロジの変更に対するコンバージェンス タイムが迅速である ( コンバージェンスはほとんど一瞬の場合もある ) 。

EIGRP は高度な距離ベクトル型プロトコルで、Diffused Update Algorithm ( DUAL ) を使用してネットワーク内の送信先への最短パスを計算します。

## プロトコルの大きな変更点

EIGRP には、バージョン 0 とバージョン 1 の大きく 2 つのバージョンがあります。Cisco IOS の 10.3(11) より前のバージョン、11.0(8)、および 11.1(3) では、旧バージョンの EIGRP が実行されます。このドキュメントの一部の説明は、旧バージョンには適用されません。EIGRP のバージョン 1 では、パフォーマンスや安定性の改良点が多数含まれるため、シスコではこちらを使用することを強くお勧めします。

## 基本的な理論

一般的な距離ベクトル型プロトコルは、送信先への最適なパスを計算するとき、距離 ( メトリックまたは距離の合計、ホップ カウントなど ) とベクトル ( ネクストホップ ) の情報を保存します。たとえば、図 1 のネットワーク内のすべてのルータは Routing Information Protocol ( RIP; ルーティング情報プロトコル ) を実行しています。ルータ 2 は、それぞれの空いているパスでホップ カウントを調べて、ネットワーク A へのパスを選択します。

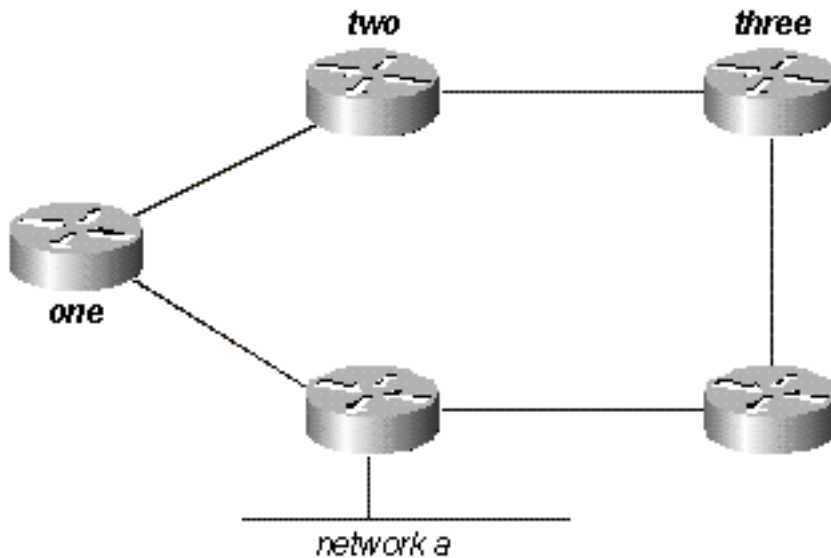


Figure 1

ルータ 3 を通るパスのホップは 3 であり、ルータ 1 を通るパスのホップは 2 であるため、ルータ 2 は ルータ 1 を通るパスを選択してルータ 3 で学習した情報を破棄します。ルータ 1 とネットワーク A の間のパスが停止すると、ルーティング テーブルのルートがタイムアウトするまで (3 アップデート期間、または 90 秒) ルータ 2 ではこの宛先とのすべての接続が不通になり、ルータ 3 はルートを再度アドバタイズします (RIP では 30 秒ごとに発生)。ルータ 2 がパスをルータ 1 からルータ 3 へ切り換えるのに、90 秒から 120 秒間かかります。これにはホールドダウン時間は含まれません。

EIGRP は、定期アップデートを使用して再コンバージする代わりに、近隣ルータの各アドバタイズメントからトポロジ テーブルを作成して (データを破棄するのではなく)、トポロジ テーブル内でループがないと思われるルートを探すか、他にルートがないことがわかっている場合は近隣ルータに問い合せて、コンバージを行います。ルータ 2 は、ルータ 1 とルータ 3 から受信した情報を保管します。ルータ 1 を通るパスを最適なパス (サクセサ) として選択し、ルータ 3 を通るパスをループのないパス (フィージブル サクセサ) として選択します。ルータ 1 を通るパスが使用できなくなった場合、ルータ 2 はトポロジ テーブルを調べて、フィージブル サクセサを見つけると、すぐにルータ 3 を通るパスを使用します。

この簡単な説明からわかるとおり、EIGRP では次の機能が提供されます。

任意時に必要なアップデートだけを送信するシステム。これは隣接ルータ検出とメンテナンスにより実現します。

ルータが学習したパスからループのないパスを調べる方法。

ネットワーク上にあるすべてのルータのトポロジ テーブルから問題のあるルートを削除するプロセス。

近隣ルータに、不通になった送信先へのパスを探すように問い合わせるプロセス。

次に、これらの各要件を順に説明します。

## 隣接ルータの検出と維持

ルーティング情報をネットワーク全体に配布するために、EIGRP は非定期的な差分ルーティング更新を使用します。つまり、EIGRP はこれらのパスが変更されたとき、変更されたパスに関するルーティング更新だけを送信します。

ルーティング更新だけを送信する場合の根本的な問題は、近接ルータを通じたパスが使用できなくなったときに、それが分からないことがあることです。新しいルーティング テーブルが隣接ルータから送信されることを期待して、ルートをタイムアウトすることはできません。EIGRP は隣接ルータとの関係を使用して、ルーティング テーブルの変更をネットワーク全体に信頼性のある方法で伝播できます。2 台のルータは、互いに相手の HELLO パケットを共通のネットワークで識別すると近隣ルータになります。

EIGRP は HELLO パケットを高帯域幅リンクでは 5 秒ごとに、低帯域幅マルチポイント リンクでは 60 秒ごとに送信します。

5 秒ごとの HELLO :

ブロードキャスト メディア (イーサネット、トークン リング、FDDI など)

ポイントツーポイント シリアル リンク (PPP や HDLC の専用回線、フレーム リレー ポイントツーポイント サブインターフェイス、および ATM ポイントツーポイント サブインターフェイス)

高帯域幅 (T1 以上) マルチポイント回線 (ISDN PRI およびフレームリレー)

60 秒ごとの HELLO :

マルチポイント回線 T1 帯域幅またはそれ以下 (フレーム リレー マルチポイント インターフェイス、ATM マルチポイント インターフェイス、ATM 相手先選択接続、および ISDN BRI)

EIGRP が HELLO パケットを送信する割合は HELLO インターバルと呼ばれ、インターフェイスごとに [ip hello-interval eigrp コマンド](#) で調整できます。待機時間は、ルータが HELLO パケットを受信しなくても近隣ルータが稼働していると思なす時間の長さです。待機時間は通常は HELLO インターバルの 3 倍で、デフォルトでは 15 秒と 180 秒です。待機時間は [ip hold-time eigrp コマンド](#) で調節できます。

HELLO インターバルを変更しても、この変更に合わせて待機時間は自動的に調節されないので注意してください。設定された HELLO インターバルを反映するために待機時間を手作業で調整する必要があります。

2 台のルータは、HELLO タイマーとホールド タイマーが一致しなくても EIGRP 隣接ルータになることができます。hello パケットには待機時間が含まれているので、各隣接ルータは hello 感覚タイマーと待機タイマーが一致しなくても稼働状態に置かれます。

ルータ上の HELLO インターバルの値を直接調べることはできませんが、隣接ルータでの [show ip eigrp neighbors](#) の出力結果から推測できます。

シスコ デバイスから [show ip eigrp neighbors](#) コマンドの出力がある場合、[アウトプット インタープリタ \(登録ユーザ専用\)](#) を使用して、潜在的な問題と修正を表示できます。アウトプットイ

ンタープリタを使用するには、JavaScript をイネーブルにする必要があります。

```
router# show ip eigrp neighborsIP-EIGRP neighbors for process 1H Address Interface Hold Uptime
SRTT RTO Q Seq Type (sec) (ms) Cnt Num1 10.1.1.2 Et1 13 12:00:53 12 300 0 6200 10.1.2.2 S0 174
12:00:56 17 200 0 645rp-2514aa# show ip eigrp neighborIP-EIGRP neighbors for process 1H Address
Interface Hold Uptime SRTT RTO Q Seq Type (sec) (ms) Cnt Num1 10.1.1.2 Et1 12 12:00:55 12 300 0
6200 10.1.2.2 S0 173 12:00:57 17 200 0 645rp-2514aa# show ip eigrp neighborIP-EIGRP neighbors
for process 1H Address Interface Hold Uptime SRTT RTO Q Seq Type (sec) (ms) Cnt Num1 10.1.1.2
Et1 11 12:00:56 12 300 0 6200 10.1.2.2 S0 172 12:00:58 17 200 0 645
```

コマンド出力の Hold カラムの値は待機時間を超えることはなく、待機時間から HELLO インターバルを引いた値より小さくなることもありません（当然、HELLO パケットがない場合を除きます）。Hold カラムの範囲が通常 10 ~ 15 秒の間であれば、HELLO インターバルは 5 秒で待機時間は 15 秒です。Hold カラムの範囲が通常 120 ~ 180 秒のように大きい場合、HELLO インターバルは 60 秒で待機時間は 180 秒です。数値がデフォルト タイマー設定値のどれにも合わない場合、近接ルータの問題のインターフェイスを調べます。HELLO タイマーとホールド タイマーが手動で設定されている場合があります。

注:

EIGRP は、ピア関係を二次アドレスで構築することはありません。すべての EIGRP トラフィックは、インターフェイスの一次アドレスを送信元としています。

マルチアクセス フレームリレー ネットワーク（ポイントツーマルチポイントなど）に EIGRP を設定する場合は、`frame-relay map` 文にキーワード `broadcast` を設定します。キーワード `broadcast` がなければ、2 つの EIGRP ルータ間に隣接関係を確立できません。[詳細は、「フレームリレーの設定とトラブルシューティング」を参照してください。](#)

EIGRP がサポートできる隣接ルータの数には制限がありません。実際にサポートされる隣接ルータの数は、次のようなデバイスの機能に依存しています。

メモリ容量

処理能力

送信されるルートの数など、交換情報の量

トポロジの複雑性

ネットワークの安定性

## [トポロジ テーブルの作成](#)

これらのルータが互いに通信状態になったとき、ルータ同士で何が交信されているのでしょうか。いうまでもなく、各ルータのトポロジ テーブルです。EIGRP は、RIP や IGRP とは異なり、ルータのルーティング（または転送）テーブルを使用して動作に必要なすべての情報を保持しません。その代わりに、第 2 の表として、トポロジ テーブルを作成し、そこからルートをルーテ

イング テーブルに挿入します。

注: Cisco IOS バージョン 12.0T および 12.1 において、RIP は専用のデータベースを保持し、そこからルートをルーティング テーブルへ挿入します。

EIGRP を実行しているルータのトポロジ テーブルの基本形式を表示するには、[show ip eigrp topology コマンド](#)を実行します。トポロジ テーブルには、到達可能な各ネットワークへの距離とベクトルの組み合わせを構築するのに必要な情報が記載されます。テーブルには次の情報があります。

上流近接ルータによってレポートされたこの送信先へのパスの最低帯域幅

遅延合計

パスの信頼性

パスの負荷

パスの最小の Maximum Transmission Unit ( MTU; 最大伝送ユニット )

フィージブル ディスタンス

レポートド ディスタンス

ルートの送信元 ( 外部ルートはマークされます )

「[到達可能距離および報告距離](#)」は、この項で後ほど説明します。

シスコ デバイスから `show ip eigrp topology` コマンドの出力がある場合、[アウトプット インタープリタ](#) ( [登録ユーザ専用](#) ) を使用して、潜在的な問題と修正を表示できます。アウトプット インタープリタを使用するには、JavaScript をイネーブルにする必要があります。

## [EIGRP メトリック](#)

EIGRP は送信先ネットワークへのパスの最小帯域幅と合計遅延を使用して、ルーティング メトリックを計算します。別のメトリックも設定できますが、ネットワーク内にルーティング ループが発生する可能性があるため、シスコではこれを推奨していません。帯域幅と遅延のメトリックは、送信先ネットワークへのパス上のルータのインターフェイスに設定された値によって求められます。

たとえば次に示す図 2 では、ルータ 1 はネットワーク A への最適なパスを計算しています。

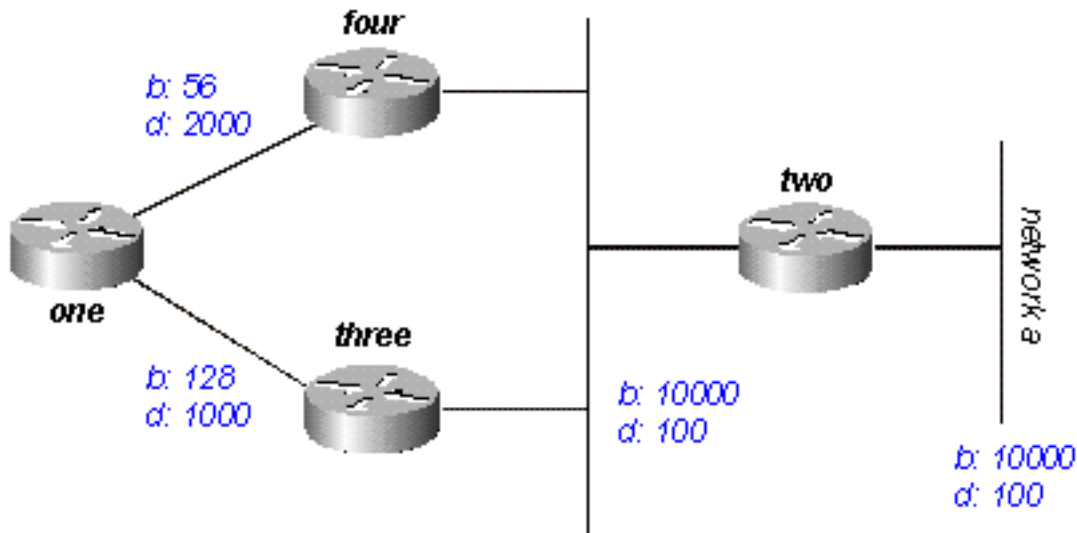


Figure 2

最初に、このネットワークの2つのアドバタイズメント ( 最小帯域幅が 56、遅延合計が 2200 でルータ 4 を通過するアドバタイズメントと、最小帯域幅が 128、遅延合計が 1200 でルータ 3 を通過するアドバタイズメント ) から始まります。ルータ 1 は、メトリックが最も低いパスを選択します。

メトリックを計算します。EIGRP は、帯域幅と遅延のメトリックをスケールして複合メトリックを計算します。EIGRP は次の数式を使用して、帯域幅をスケールします。

$$\text{帯域幅} = ( 10000000 / \text{帯域幅} ( i ) ) * 256$$

ここで、帯域幅 ( i ) は、宛先ネットワークへのルート上のすべての発信インターフェースの最小帯域幅で、キロビットで表されます。

EIGRP は次の数式を使用して、遅延をスケールします。

$$\text{遅延} = \text{遅延} ( i ) * 256$$

ここで、遅延 ( i ) は、宛先ネットワークへのルート上のインターフェースで設定されている遅延の合計で、10 マイクロ秒によって表されます。show ip eigrp topology や show interface コマンドで表示される遅延はマイクロ秒単位であるため、10 で割ってからこの数式で使用する必要があります。この文書の中では、インターフェースに設定され表示されている遅延を使用します。

EIGRP は、これらのスケールされた値を使用して、ネットワークへの複合メトリックを求めます。

$$\text{メトリック} = ([K1 * \text{帯域幅} + ( K2 * \text{帯域幅} ) / ( 256 \text{ 負荷} ) + K3 * \text{遅延}] * [K5 / ( \text{信頼性} + K4 ) ]) * 256$$

注: これらの K の値は、慎重に計画して使用する必要があります。K の値のミスマッチによって、近隣関係を作ることができず、ネットワークがコンバートできない原因になります。

注: K5 = 0 の場合、式は  $\text{メトリック} = ([k1 * \text{帯域幅} + (k2 * \text{帯域幅}) / (256 - \text{負荷}) + k3 * \text{遅延}) * 256$  になります。

K のデフォルト値は、次のとおりです。

$$K1 = 1$$

$$K2 = 0$$

$$K3 = 1$$

$$K4 = 0$$

$$K5 = 0$$

デフォルトの動作の場合、数式を次のように簡単にできます。

`metric = bandwidth + delay`

シスコのルータは浮動小数点計算をしないため、計算の各段階でメトリックを正しく計算するには、小数点以下を切り捨てる必要があります。この例では、ルータ 4 を通る合計コストは、次のとおりです。

この例では、ルータ 4 を通る合計コストは、次のとおりです。

`minimum bandwidth = 56k total delay = 100 + 100 + 2000 = 2200 [(10000000/56) + 2200] x 256 = (178571 + 2200) x 256 = 180771 x 256 = 46277376`

さらに、ルータ 3 を通る合計コストは次のとおりです。

`minimum bandwidth = 128k total delay = 100 + 100 + 1000 = 1200 [(10000000/128) + 1200] x 256 = (78125 + 1200) x 256 = 79325 x 256 = 20307200`

したがってネットワーク A に到達するために、ルータ 1 はルータ 3 を通るルートを選択します。

ここで使用した帯域幅と遅延の値は、ルータが宛先ネットワークへのネクストホップへ到達するために通るインターフェイスに設定した値であることに注意してください。たとえばルータ 2 が、イーサネット インターフェイスで設定されている遅延でネットワーク A にアドバタイズするとします。ルータ 4 は自分のイーサネット で設定された遅延を追加し、ルータ 1 は自分のシリアルで設定された遅延を追加しました。

## 到達可能距離、報告距離、およびフィージブル サクセサ

到達可能距離は送信先ネットワークへのパスの最適なメトリックであり、このパスをアドバタイズしている近隣ルータのメトリックが含まれます。報告距離は、上流近接ルータによってアドバタイズされる送信先ネットワークへのパスの複合メトリックです。フィージブル サクセサは、報告距離 (現在のベスト パス) が到達可能距離より小さなパスです。図 3 では、このプロセスを説明します。



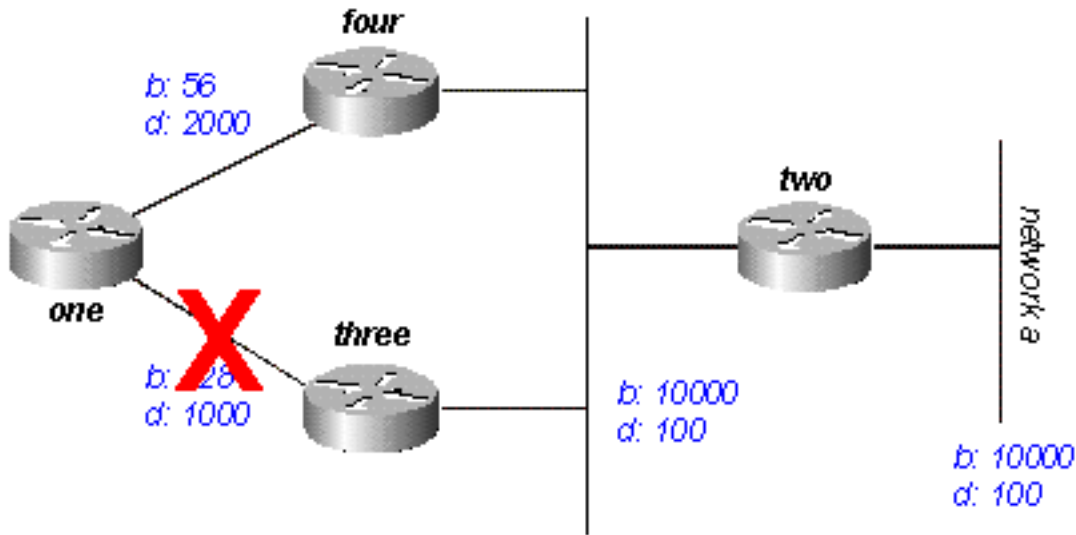


Figure 3

ルータ 1 は、ネットワーク A へのパスが 2 つあることを認識します。1 つはルータ 3 を通り、もう 1 つはルータ 4 を通ります。

ルータ 4 を通るルートのコストは 46277376 で、報告距離は 307200 です。

ルータ 3 を通るルートのコストは 20307200 で、報告距離は 307200 です。

それぞれのケースで EIGRP は、ネットワークへのルートをアドバタイズしているルータからの報告距離を計算することに注意してください。つまり、ルータ 4 からの報告距離はルータ 4 からネットワーク A へ到達するメトリックで、ルータ 3 からの報告距離はルータ 3 からネットワーク A に到達するメトリックになります。EIGRP はルータ 3 を通るルートを最適なパスに選択し、ルータ 3 を通るメトリックを到達可能距離として使用します。このネットワークへのルータ 4 を通る報告距離は到達可能距離より小さいため、ルータ 1 はルータ 4 を通るパスをフェイジブルサクセサと見なします。

ルータ 1 とルータ 3 の間のリンクが停止すると、ルータ 1 は把握しているネットワーク A への各パスを調べてルータ 4 にフェイジブルサクセサがあることを見つけます。ルータ 1 はこのルートを使用し、ルータ 4 を通るメトリックを新しい到達可能距離として使用します。ネットワークはただちにコンバージして、下流近接ルータのアップデートがルーティングプロトコルからの唯一のトラフィックになります。

図 4 では、もう少し複雑なシナリオを説明します。

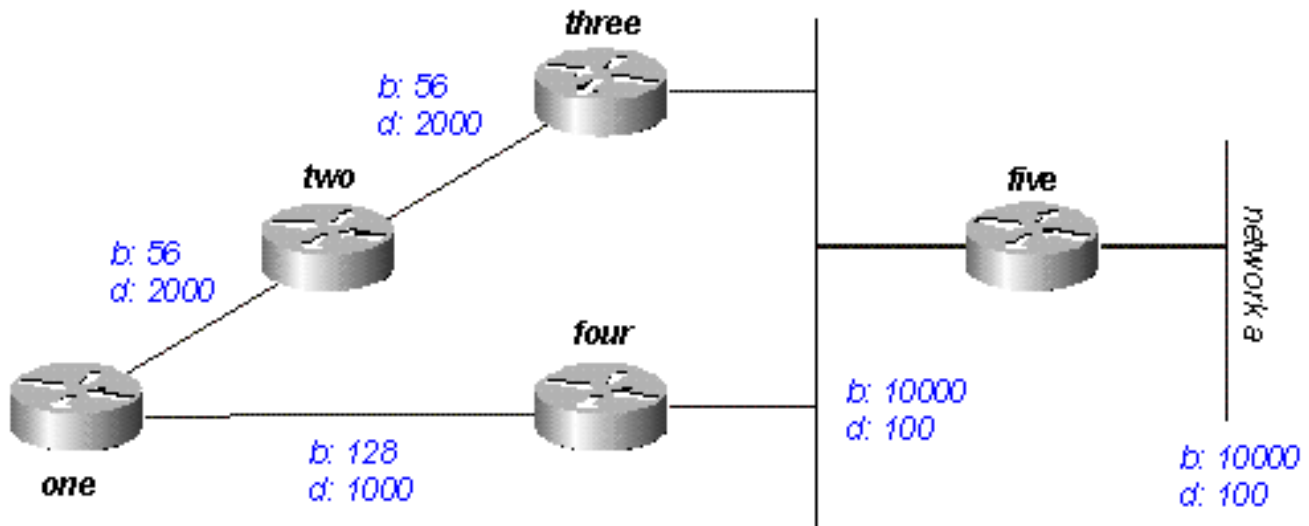


Figure 4

ルータ 1 からネットワーク A へのルートは 2 つあります。これは、ルータ 2 を通りメトリックが 46789376 のルートと、ルータ 4 を通りメトリックが 20307200 のルートです。ルータ 1 はこの 2 つのメトリックで低い方をネットワーク A へのルートに選択し、このメトリックが到達可能距離になります。次に、ルータ 2 を通るパスを見てフィージブルサクセサにできるかどうかを調べます。ルータ 2 からの報告距離は 46277376 でこれは到達可能距離の値より高いため、このパスはフィージブルサクセサではありません。この時点でルータ 1 のトポロジテーブルを見ると ( `show ip eigrp topology` を使用 )、ルータ 4 を通るネットワーク A のエントリが 1 つだけ表示されているのが分かります。( 実際には、ルータ 1 のトポロジテーブルには 2 つのエントリがありますが、フィージブルサクセサであるのは 1 つだけであるため、もう 1 つのエントリは `show ip eigrp topology` では表示されません。フィージブルサクセサではないルートは `show ip eigrp topology all-links` を使用すると表示できます )。

ルータ 1 とルータ 4 の間のリンクが停止していると想定しましょう。ルータ 1 はネットワーク A への唯一のルートが不通になったことを認識し、各近隣ルータ ( この場合はルータ 2 だけ ) に問い合わせネットワーク A へのルートがあるかどうかを調べます。ルータ 2 にはネットワーク A へのルートがあるため、クエリーに応答します。ルータ 1 はルータ 4 を通る適切なルートがないため、ルータ 2 を通るネットワーク A へのこのルートを受け入れます。

### パスにループがないかどうかを判断する

EIGRP は到達可能距離、報告距離、フィージブルサクセサの概念をどう使用して、パスが有効か、ループがないかを判断するのでしょうか。図 4a では、ルータ 3 はネットワーク A へのルートを調べます。スプリットホライズンがディセーブルになっているため (たとえば、マルチポイントフレームリレーインターフェイスがある場合など)、ルータ 3 からネットワーク A へのルートは 3 つあります。ルータ 4 を通るパス、ルータ 2 を通るパス (パスは 2、1、3、4)、およびルータ 1 を通るパス (パスは 1、2、3、4) の 3 つを表示します。

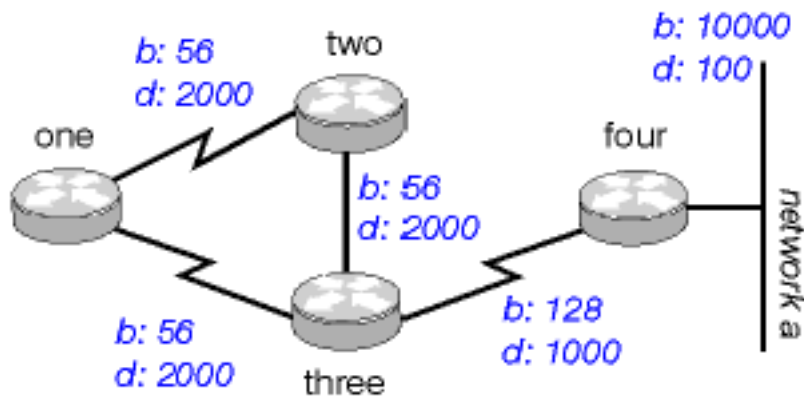


Figure 4a

ルータ 3 がこれらすべてのルートを受け入れると、ルーティング ループが発生します。ルータ 3 はルータ 2 を通ってネットワーク A に到達できると考えますが、ルータ 2 を通るパスがネットワーク A へ到達するにはルータ 3 を通ります。ルータ 4 とルータ 3 の接続が停止すると、ルータ 3 は他のパスを通ればネットワーク A へ到達できると考えますが、フィージブル サクセサを決定するルールがあるため、これらのパスを代わりに使用することはありません。メトリックで、その理由を調べましょう。

ルータ 4 を通るネットワーク A への複合メトリック： 20281600

ルータ 2 を通るネットワーク A への複合メトリック： 47019776

ルータ 1 を通るネットワーク A への複合メトリック： 47019776

ルータ 4 を通るパスは最適なメトリックであるため、ルータ 3 はこのルートを転送テーブルに設定し、20281600 をネットワーク A への到達可能距離として使用します。ルータ 3 は、報告されたルータ 2 と 1 を経由したネットワーク A までの距離を計算します。47019776 はルータ 2 を通るパスで、47019776 はルータ 1 を通るパスです。これらのメトリックはどちらも到達可能距離より大きいため、ルータ 3 はどちらのルートもネットワーク A のフィージブル サクセサとして設定しません。

ルータ 3 とルータ 4 の間のリンクが停止していると仮定します。ルータ 3 はその隣接ルータに対し、ネットワーク A までの代替パスを問い合わせます。ルータ 2 はクエリーを受信し、このクエリーがサクセサから送信されているため、トポロジ テーブル内の他の各エントリを検索してフィージブル サクセサがあるかどうかを確認します。トポロジ テーブル内にある他の唯一のエントリはルータ 1 からのもので、報告距離はルータ 3 を通る最後の既知の最適メトリックと等しくなります。ルータ 1 を通る報告距離は最後の既知の到達可能距離より小さくないため、ルータ 2 はこのルートを到達不能にマークして各近隣ルータ (この場合はルータ 1 だけ) にネットワーク A へのパスを問い合わせます。

ルータ 3 はネットワーク A のクエリーをルータ 1 にも送信します。ルータ 1 は自分のトポロジ テーブルを調べて、ネットワーク A への他の唯一のパスが、ルータ 2 を通るものであり、その報告距離がルータ 3 を通る最後の既知の到達可能距離と等しいことが分かります。もう一度繰り返しますが、ルータ 2 を通る報告距離は最後の既知の到達可能距離より小さくないため、このルートはフィージブル サクセサにはなりません。ルータ 1 はこのルートを到達不能にマークして、他の唯一の近隣ルータであるルータ 2 にネットワーク A へのパスを問い合わせます。

これは第一段階のクエリーです。ルータ 3 は、ネットワーク A へのルートを検出するため各隣接ルータに対してクエリーを実行しました。次に、ルータ 1 とルータ 2 はルートを到達不能とし

てマークして、ネットワーク A へのルートを検出するため残りの各隣接ルータに対してクエリーを実行しました。ルータ 2 はルータ 1 のクエリーを受信すると、自分のトポロジ テーブルを調べて送信先が到達不能にマークされていることを確認します。ルータ 2 はルータ 1 にネットワーク A は到達不能であると応答します。ルータ 1 もまた、ルータ 2 のクエリーを受信すると、ネットワーク A が到達不能であると応答します。これでルータ 1 と 2 は両方ともネットワーク A は到達不能であると判断し、ルータ 3 の最初のクエリーに応答します。ネットワークはコンバージして、すべてのルートはパッシブ状態に戻ります。

## スプリット ホライズンとポイズン リバース

前の例では、スプリット ホライズンが有効でない場合を想定し、ルートがループしているかどうかを判断するために EIGRP が実行可能な距離および報告距離をどのように使用するかを示しました。しかし、状況によっては、EIGRP はスプリット ホライズンを使用して同様にルーティング ループを回避します。EIGRP におけるスプリット ホライズンの使用方法について詳しく説明する前に、スプリット ホライズンとは何か、どう動作するのかを説明します。スプリット ホライズンの規則は次のとおりです。

ルートを学習したインターフェイスから同じルートをアドバタイズしない。

たとえば、図 4a でルータ 1 が単一のマルチポイント インターフェイス ( フレームリレーなど ) を通じてルータ 2 とルータ 3 に接続しているとします。ここでルータ 1 がネットワーク A についてルータ 2 から学習した場合、ルータ 1 は同じインターフェイスからネットワーク A へのルートをルータ 3 にアドバタイズしません。ルータ 1 は、ルータ 3 がネットワーク A についてルータ 2 から直接学習すると想定します。

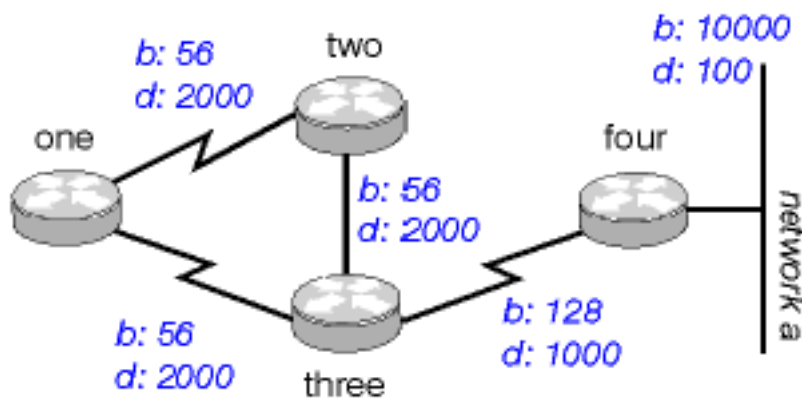


Figure 4a

ポイズン リバースはルーティング ループを回避するもう 1 つの方法です。ポイズン リバースの規則は次のとおりです。

インターフェイスを通じてルートを学習したら、同じインターフェイスから、そのルートを到達不能としてアドバタイズする。

図 4a のルータでポイズン リバースが有効であるとします。ルータ 1 はネットワーク A についてルータ 2 から学習すると、ルータ 2 およびルータ 3 へのリンクを通じてネットワーク A を到達不能としてアドバタイズします。ルータ 3 がルータ 1 を経由したネットワーク A へのパスを保持している場合、ルータ 3 は到達不能アドバタイズメントを受信したことでそのパスを削除します。EIGRP はこれら 2 つの規則を組み合わせることでルーティング ループを回避します。

EIGRP は次のような場面でスプリット ホライズンを使用するか、またはルートを到達不能としてアドバタイズします。

2 台のルータが起動モードにある ( トポロジ テーブルを初めて交換する ) 。

トポロジ テーブルの変更をアドバタイズする。

クエリーを送信する。

それぞれの状況について検討します。

## 起動モード

2 台のルータが初めて近接ルータとなる場合、両ルータは起動モード中にトポロジ テーブルを交換します。ルータは起動モード中に受信したテーブル エントリごとに、同じエントリを最大メトリックで新しい近接ルータにアドバタイズし戻します ( ポイズン ルート ) 。

## トポロジ テーブルの変更

図 5 で、ルータ 1 は 2 つのシリアル リンク ( ルータ 2 とルータ 4 を接続する 56k リンクと、ルータ 3 とルータ 4 を接続する 128k リンク ) の間でネットワーク A 宛てのトラフィックのバランスをとるためにバリエーションを使用します ( バリエーションに関する説明については「[ロード バランシング](#)」の項を参照 ) 。

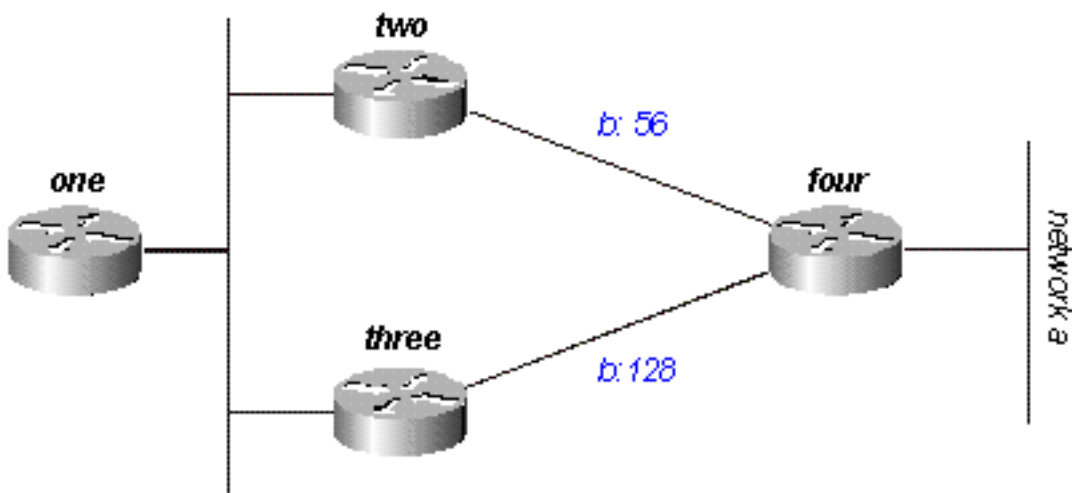


Figure 5

ルータ 2 はルータ 3 を経由するパスを、実行可能な後継ルータと見なしています。ルータ 2 とルータ 4 の間のリンクがダウンした場合、ルータ 2 は単にルータ 3 を経由するパス上で再コンバートします。スプリット ホライズン規則により、ルートを学習したインターフェイスから同じルートがアドバタイズされないため、通常、ルータ 2 はアップデートを送信しません。ただし、それではルータ 1 のトポロジ テーブルに無効なエントリが残ります。ルータのトポロジ テーブル内で、ルータがあるネットワークに到達する際に経由するインターフェイスが変わった場合、スプリット ホライズンはオフになり、古いルートがすべてのインターフェイスからポイズン リバースされます。このケースでは、ルータ 2 はこのルートのスプリット ホライズンをオフにし、ネットワーク A を到達不能としてアドバタイズします。ルータ 1 はこのアドバタイズメントを受

信し、ルータ 2 を経由してネットワーク A に至るルートルーティングテーブルからフラッシュします。

## クエリー

クエリーによってスプリットホライズンが発生するのは、クエリー内で宛先へのサクセサとして使用しているルータからクエリーまたはアップデートを受信した場合だけです。図 6 のネットワークを参照してください。

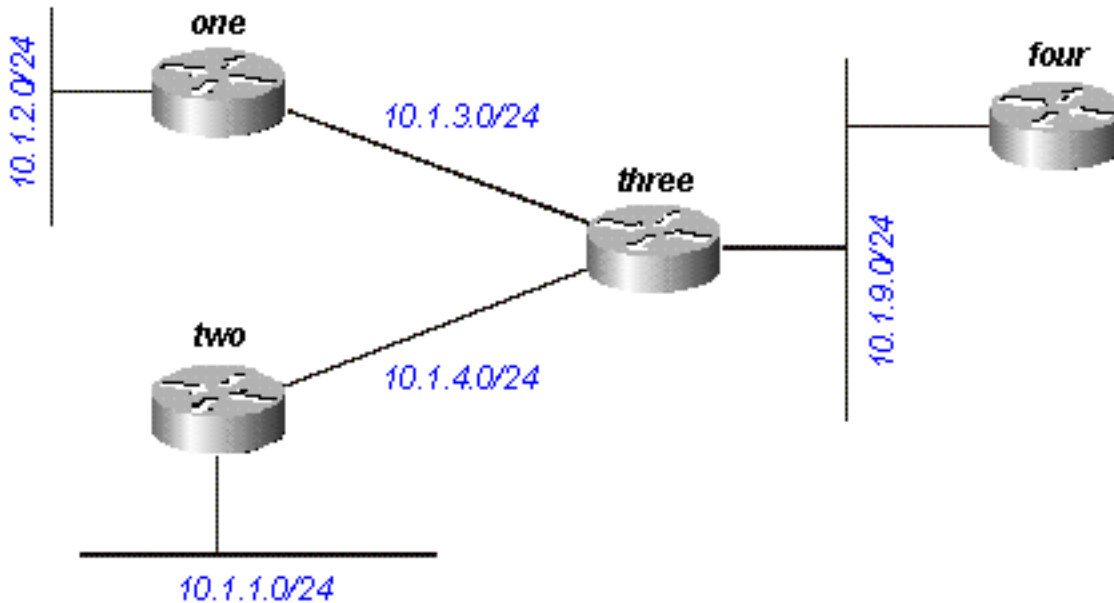


Figure 6

ルータ 3 はルータ 4 から 10.1.2.0/24 (ルータ 3 はルータ 1 を経由してこのネットワークに到達する) に関するクエリーを受信します。リンクフラップまたはその他の一時的なネットワーク状態が原因でルータ 3 にこの宛先のサクセサがない場合、ルータ 3 は各隣接ルータにクエリーを送信します。このケースではルータ 1、2、および 4 ただし、ルータ 3 がルータ 1 から宛先 10.1.2.0/24 に関するクエリーまたはアップデート (メトリックの変更など) を受信した場合、ルータ 3 はルータ 1 にクエリーを送信しません。これは、ルータ 3 にとってルータ 1 は 10.1.2.0/24 への後継ルータであるためです。その結果、ルータ 3 はルータ 2 とルータ 4 にだけクエリーを送信します。

## Stuck In Active ルート

状況によっては、クエリーの応答に非常に時間がかかる場合があります。実際に長時間応答がなければ、クエリーを発行したルータは応答の回収を中止し、応答が得られなかったルータとの接続を解除して、その近接ルータとのセッションを事実上再起動します。これを Stuck In Active (SIA) ルートと呼びます。最も基本的な SIA ルートは、クエリーがネットワークの反対側に到達して応答が戻ってくるまでに単に時間がかかりすぎる場合に発生します。たとえば、図 7 でルータ 1 はルータ 2 から数多くの SIA ルートを記録しているとします。

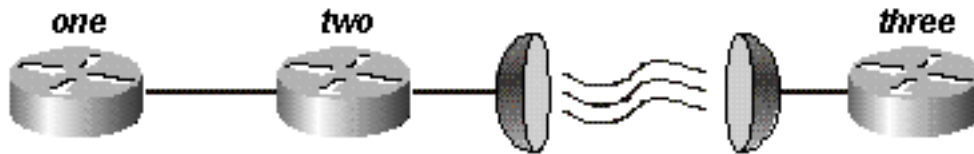


Figure 7

調査の結果、問題はルータ 2 とルータ 3 を接続する衛星リンク上の遅延にあることがわかりました。この種の問題には 2 通りの解決策があります。1 つは、ルータがクエリーを送信してから SIA ルートを宣言するまでの時間を長くすることです。この設定を変更するには、[timers active-time コマンド](#)を使用します。

しかし、さらに優れた解決策として、クエリーの範囲が狭くなるようにネットワークを再設計する方法があります (つまり、ごく小数のクエリーしか衛星リンクを通過しない)。クエリーの範囲については、「[クエリーの処理と範囲](#)」の項で説明します。ただし、クエリー範囲自体は、SIA ルートが報告される一般的な原因ではありません。よく見られるのは、ネットワーク上の一部のルータが次のいずれかの原因でクエリーに応答できない場合です。

ルータが過度のビジー状態にあり、クエリーに応答できない (CPU 高使用率が発生しているケースが多い)。

ルータのメモリに問題があり、クエリーの処理または応答パケットの組み立てにメモリを割り当てることができない。

2 台のルータ間の回線が良好な状態にない。近接ルータとの関係を維持できるだけのパケットは流れていても、一部のクエリーや応答がルータ間で失われている。

単方向リンク (障害のためにトラフィックが一方向にしか流れないリンク)。

## [SIA ルートのトラブルシューティング](#)

SIA ルートのトラブルシューティングには一般に 3 つのステップがあります。

常に SIA として報告されるルートを特定する。

これらのルートに関するクエリーにまったく応答していないルータを特定する。

そのルータがクエリーを受信できない、またはクエリーに応答できない原因を特定する。

最初のステップは非常に簡単です。コンソールメッセージをログに記録していれば、ログにざっと目を通すだけで、頻繁に SIA としてマークされているルートがわかります。2 番目のステップは 1 番目のステップほど簡単ではありません。この情報を収集するコマンドは `show ip eigrp topology active` です。

```
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,          r - Reply status A
10.2.4.0/24, 0 successors, FD is 512640000, Q      1 replies, active 00:00:01, query-origin:
```

```
Local origin          via 10.1.2.2 (Infinity/Infinity), Serial1    1 replies, active 00:00:01,
query-origin: Local origin          via 10.1.3.2 (Infinity/Infinity), r, Serial3    Remaining
replies:              via 10.1.1.2, r, Serial0
```

まだ応答していない隣接ルータは R で示されます (アクティブ タイマーによって、そのルートがどれくらいの時間アクティブであるかがわかります)。これらの隣接ルータには Remaining replies セクションが表示されない場合があります。これらは他の RDB と共に表示される可能性があります。未受信の応答があるルートで、ある一定の時間 (通常は 2 ~ 3 分) アクティブであるルートには特に注意してください。このコマンドを複数回実行すれば、どの近接ルータがクエリーに回答していないか (または、どのインターフェイスに未応答のクエリーが多いか) がわかります。この隣接ルータを調べて、そのルータが自身の隣接ルータのいずれかからの応答を待ち続けているのかどうかを確認します。このプロセスを繰り返して、クエリーにまったく応答していないルータを特定します。ルータが特定されたら、その近接ルータへのリンクや、メモリまたは CPU の使用率などに問題がないかを調べます。

クエリー範囲が問題であるように思われる場合は、SIA タイマーを増やすのではなく、クエリー範囲を狭くするのが常に最良の方法です。

## 再配布

この項では、再配送に関するいくつかの異なるシナリオを検討します。ただし、以下に示すのは、再配送の設定に必要とされる最低限の例であることに注意してください。再配送によって、最適ではないルーティング、ルーティング ループ、コンバージェンス遅延などの問題が発生する可能性があります。これらの問題を回避するには、「ルーティング プロトコルの再配送」の「再配送による障害の回避」を参照してください。

### 2 つの EIGRP 自律システム (AS) 間の再配布

図 8 ではルータは次のように設定されています。

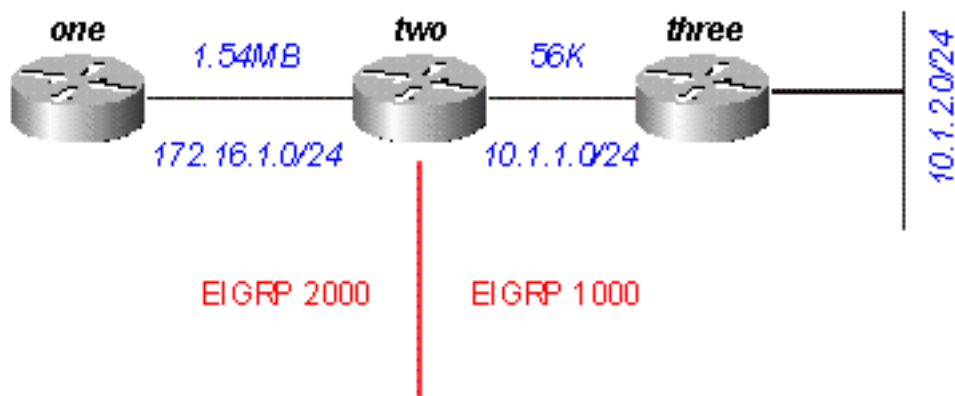


Figure 8

#### ルータ 1

```
router eigrp 2000 !--- The "2000" is the autonomous system network 172.16.1.0 0.0.0.255
```

#### ルータ 2

```
router eigrp 2000 redistribute eigrp 1000 route-map to-eigrp2000 network 172.16.1.0 0.0.0.255 !
router eigrp 1000 redistribute eigrp 2000 route-map to-eigrp1000 network 10.1.0.0
0.0.255.255route-map to-eigrp1000 deny 10match tag 1000!route-map to-eigrp1000 permit 20set tag
2000!route-map to-eigrp2000 deny 10match tag 2000!route-map to-eigrp2000 permit 20set tag 1000
```



## ルータ 3

```
router eigrp 1000 network 10.1.0.0 0.0.255.255
```

ルータ 3 は、自律システム 1000 を介してルータ 2 にネットワーク 10.1.2.0/24 をアドバタイズします。ルータ 2 はこのルートを実 AS 2000 に再配送して、ルータ 1 にアドバタイズします。

注: EIGRP 1000 からのルートは、EIGRP 2000 に再配布される前に 1000 がタグとして追加されます。EIGRP 2000 からのルートが EIGRP 1000 に再配布される時に、ループなしのトポロジを保証するため、1000 タグが付けられたルートが拒否されます。ルーティングプロトコルの再配送の詳細は、『[ルーティングプロトコルの再配布](#)』を参照してください。

ルータ 1 には次のように表示されます。

```
one# show ip eigrp topology 10.1.2.0 255.255.255.0 IP-EIGRP topology entry for 10.1.2.0/24 State is Passive, Query origin flag is 1, 1 Successor(s), FD is 46763776 Routing Descriptor Blocks: 20.1.1.1 (Serial0), from 20.1.1.1, Send flag is 0x0 Composite metric is (46763776/46251776), Route is External Vector metric: Minimum bandwidth is 56 Kbit Total delay is 41000 microseconds Reliability is 255/255 Load is 1/255 Minimum MTU is 1500 Hop count is 2 External data: Originating router is 10.1.2.1 AS number of route is 1000 External protocol is EIGRP, external metric is 46251776 Administrator tag is 1000 (0x000003E8)
```

ルータ 1 とルータ 2 間のリンクの帯域幅が 1.544Mb であるにもかかわらず、このトポロジテーブル エントリに表示される最低帯域幅は 56k であることに注意してください。このことから、EIGRP が 2 つの EIGRP AS 間で再配送を行う際には、すべてのメトリックを保存していることがわかります。

## 2 つの異なる AS にある EIGRP と IGRP 間の再配布

図 9 では、設定を次のように変更しています。

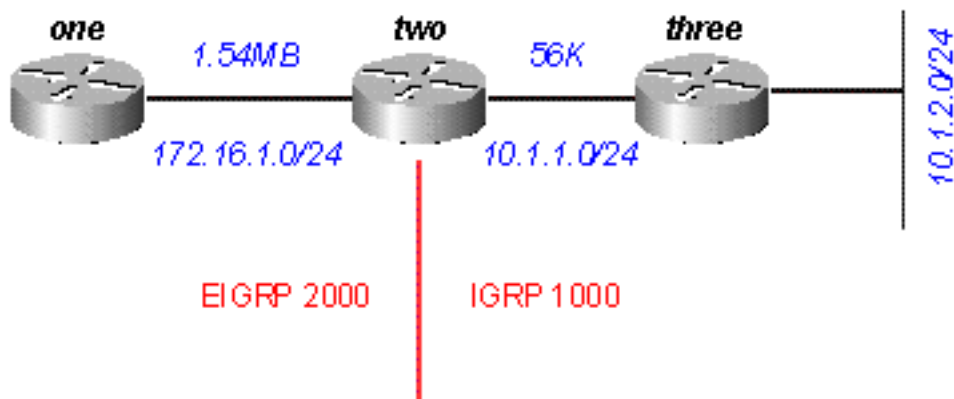


Figure 9

## ルータ 1

```
router eigrp 2000 network 172.16.1.0
```

## ルータ 2

```
router eigrp 2000 redistribute igrp 1000 route-map to-eigrp2000 network 172.16.1.0! router igrp 1000 redistribute eigrp 2000 route-map to-igrp1000 network 10.0.0.0 !route-map to-igrp1000 deny 10match tag 1000!route-map to-igrp1000 permit 20set tag 2000!route-map to-eigrp2000 deny 10match tag 2000!route-map to-eigrp2000 permit 20set tag 1000
```

### ルータ 3

```
router igrp 1000 network 10.0.0.0
```

ルータ 1 の設定を次に示します。

```
one# show ip eigrp topology 10.1.2.0 255.255.255.0 IP-EIGRP topology entry for 10.1.2.0/24 State is Passive, Query origin flag is 1, 1 Successor(s), FD is 46763776 Routing Descriptor Blocks: 20.1.1.1 (Serial0), from 20.1.1.1, Send flag is 0x0 Composite metric is (46763776/46251776), Route is External Vector metric: Minimum bandwidth is 56 Kbit Total delay is 41000 microseconds Reliability is 255/255 Load is 1/255 Minimum MTU is 1500 Hop count is 1 External data: Originating router is 10.1.1.1 AS number of route is 1000 External protocol is IGRP, external metric is 180671 Administrator tag is 1000 (0x000003E8)
```

IGRP メトリックは、ルートが別の AS を持つ EIGRP 内に再配送された場合は保存されますが、IGRP メトリックに定数 256 を掛けることでスケールされます。IGRP と EIGRP 間の再配送に関して、注意点が 1 つあります。再配送を実行しているルータにネットワークが隣接している場合、このネットワークはメトリック 1 でルートをアドバタイズします。

たとえば、ネットワーク 10.1.1.0/24 がルータ 2 に隣接していて、IGRP がこのネットワークにルーティングしているとします (ルータ IGRP の下に、このインターフェイスについて述べたネットワーク ステートメントがあります)。EIGRP はこのネットワークのルーティングプロトコルではないので、IGRP からの再配送を介して、この隣接したインターフェイスについて学習します。ルータ 1 では、10.1.1.0/24 のトポロジ テーブル エントリが次のように表示されます。

```
one# show ip eigrp topology 10.1.1.0 255.255.255.0 IP-EIGRP topology entry for 10.1.1.0/24 State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2169856 Routing Descriptor Blocks: 20.1.1.1 (Serial0), from 20.1.1.1, Send flag is 0x0 Composite metric is (2169856/1), Route is External Vector metric: Minimum bandwidth is 1544 Kbit Total delay is 20000 microseconds Reliability is 0/255 Load is 1/255 Minimum MTU is 1500 Hop count is 1 External data: Originating router is 10.1.1.1 AS number of route is 1000 External protocol is IGRP, external metric is 0 Administrator tag is 1000 (0x000003E8)
```

太字で示される、ルータ 2 からの報告距離が 1 インチ ( 2.54 cm ) であることに注意してください。

### 同じ AS にある EIGRP と IGRP 間の再配布

図 10 では、ルータ設定には、次の変更が追加されています。

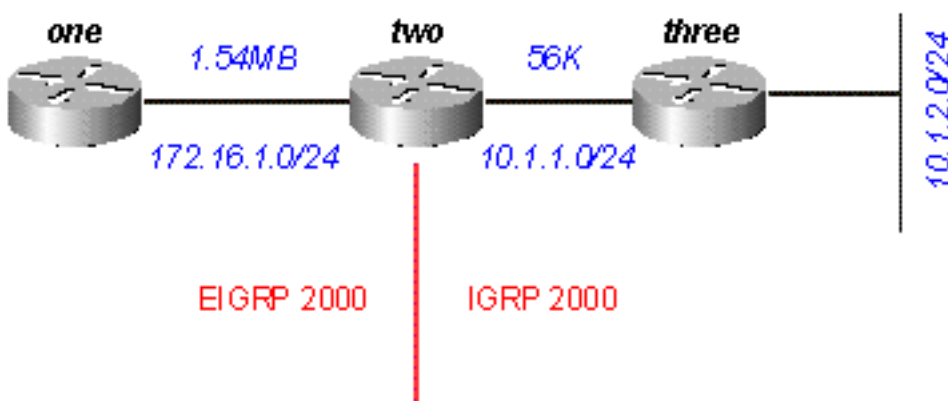


Figure 10

## ルータ 1

```
router eigrp 2000 network 172.16.1.0
```

## ルータ 2

```
router eigrp 2000 network 172.16.1.0! router igrp 2000 network 10.0.0.0
```

## ルータ 3

```
router igrp 2000 network 10.0.0.0
```

ルータ 1 は次のように設定されます。

```
one# show ip eigrp topology 10.1.2.0 255.255.255.0 IP-EIGRP topology entry for 10.1.2.0/24 State is Passive, Query origin flag is 1, 1 Successor(s), FD is 46763776 Routing Descriptor Blocks: 20.1.1.1 (Serial0), from 20.1.1.1, Send flag is 0x0 Composite metric is (46763776/46251776), Route is External Vector metric: Minimum bandwidth is 56 Kbit Total delay is 41000 microseconds Reliability is 255/255 Load is 1/255 Minimum MTU is 1500 Hop count is 1 External data: Originating router is 10.1.1.1 AS number of route is 2000 External protocol is IGRP, external metric is 180671 Administrator tag is 0 (0x00000000)
```

この設定は、前述した、IGRP と EIGRP を実行する 2 つの異なる AS 間の再配布の出力とよく似ています。隣接する 10.1.1.0/24 ネットワークは、両シナリオで同じように処理されます。

```
one# show ip eigrp topology 10.1.1.0 255.255.255.0 IP-EIGRP topology entry for 10.1.1.0/24 State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2169856 Routing Descriptor Blocks: 20.1.1.1 (Serial0), from 20.1.1.1, Send flag is 0x0 Composite metric is (2169856/1), Route is External Vector metric: Minimum bandwidth is 1544 Kbit Total delay is 20000 microseconds Reliability is 255/255 Load is 1/255 Minimum MTU is 1500 Hop count is 1 External data: Originating router is 10.1.1.1 AS number of route is 2000 External protocol is IGRP, external metric is 0 Administrator tag is 0 (0x00000000)
```

そのため、ルータ 1 と隣接するこのネットワークは、IGRP から EIGRP に、1 のメトリックで再配布されます。これは、2 つの異なる AS 間で再配布を行う場合と同じメトリックです。

同じ AS 内での EIGRP/IGRP 再配送に関して、注意点が 2 つあります。

内部 EIGRP ルートは、外部 EIGRP/IGRP ルートよりも常に優先されます。

外部 EIGRP ルートのメトリックは、スケールされた IGRP メトリックと比較されます ( 管理上の距離は無視されます ) 。

これらの注意点について、図 11 で検証します。

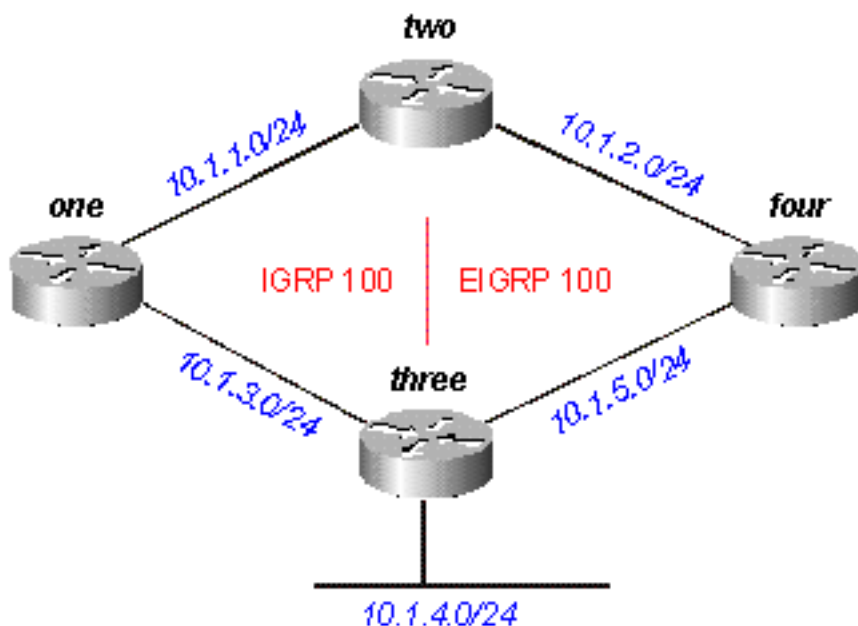


Figure 11

ルータ 1 は、IGRP 自律システム 100 で 10.1.4.0/24 をアドバタイズします。ルータ 4 は、EIGRP 自律システム 100 で 10.1.4.0/24 を外部としてアドバタイズします。ルータ 2 は AS 100 で、EIGRP と IGRP を両方とも動作させます。

ルータ 4 によってアドバタイズされる EIGRP ルートを見捨てた場合 (たとえばルータ 2 とルータ 4 間のリンクをシャットダウンするなどして)、ルータ 2 に次のように表示されます。

```
two# show ip route 10.1.4.0 Routing entry for 10.1.4.0/24 Known via "igrp 100", distance 100,
metric 12001 Redistributing via igrp 100, eigrp 100 Advertised by igrp 100 (self originated)
eigrp 100 Last update from 10.1.1.2 on Serial1, 00:00:42 ago Routing Descriptor Blocks: *
10.1.1.2, from 10.1.1.2, 00:00:42 ago, via Serial1 Route metric is 12001, traffic share count is
1 Total delay is 20010 microseconds, minimum bandwidth is 1000 Kbit Reliability 1/255, minimum
MTU 1 bytes Loading 1/255, Hops 0
```

アドミニストレーティブ ディスタンスは 100 であることに注意してください。EIGRP ルートを追加すると、ルータ 2 に次のように表示されます

```
two# show ip route 10.1.4.0 Routing entry for 10.1.4.0/24 Known via "eigrp 100", distance 170,
metric 3072256, type external Redistributing via igrp 100, eigrp 100 Last update from 10.1.2.2
on Serial0, 00:53:59 ago Routing Descriptor Blocks: * 10.1.2.2, from 10.1.2.2, 00:53:59 ago, via
Serial0 Route metric is 3072256, traffic share count is 1 Total delay is 20010 microseconds,
minimum bandwidth is 1000 Kbit Reliability 1/255, minimum MTU 1 bytes Loading 1/255, Hops 1
```

これら 2 つのルートのメトリックは、IGRP から EIGRP にスケールされた後、同じになります (「[メトリックの使用](#)」の項を参照してください)。

$$12001 \times 256 = 3072256$$

ここで、12001 (IGRP メトリック) はルータ 1 を通過し、3072256 (EIGRP メトリック) はルータ 4 を通過します。

ルータ 2 は、(スケール後の) 同じメトリックの EIGRP 外部ルートと、大きい方のアドミ

ニストレーティブ ディスタンスを優先させます。同じ AS にある EIGRP と IGRP 間で自動再配送が行われた場合は、常にそうなります。ルータは常に、最低のコスト メトリックのパスを優先させ、管理上の距離は無視します。

## その他のプロトコルとの間の再配布

EIGRP と、その他のプロトコル (たとえば RIP、OSPF など) との間の再配送は、すべての再配送と同じように行われます。プロトコル間の再配布では、デフォルト メトリックを使用することが常に理想的です。EIGRP とその他のプロトコルの間で再配布を行う場合は、次の 2 つの問題に注意する必要があります。

EIGRP に再配布されるルートは、常に集約されるわけではありません。「[集約](#)」の項で説明を参照してください。

外部 EIGRP ルートでは、管理上の距離は 170 です。

## インターフェイスへのスタティックルートの再配布

スタティック ルートをインターフェイスに設定して、このスタティック ルートを含む **router eigrp** を使ってネットワーク ステートメントを設定する場合、EIGRP はルートが直接インターフェイスに接続しているかのように、このルートを再配布します。図 12 のネットワークを参照してください。

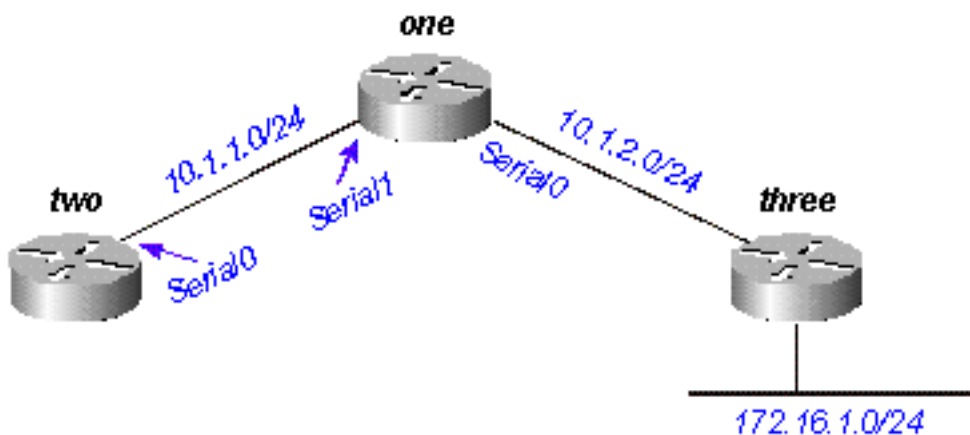


Figure 12

ルータ 1 には、インターフェイス シリアル 0 を介して設定されたネットワーク 172.16.1.0/24 へのスタティック ルートがあります。

```
ip route 172.16.1.0 255.255.255.0 Serial0
```

そしてルータ 1 には、このスタティック ルートの宛先に向けられたネットワーク ステートメントもあります。

```
router eigrp 2000 network 10.0.0.0 network 172.16.0.0 no auto-summary
```

ルータ 1 はスタティック ルートを再配送しないにもかかわらず、このルートは再配送します。これは、EIGRP がこのルートを、隣接したネットワークとみなすためです。ルータ 2 では次のように表示されます。

```
two# show ip route .... 10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks C 10.1.1.0/24 is
```

directly connected, Serial0 D 10.1.2.0/24 [90/2169856] via 10.1.1.1, 00:00:47, Serial0  
172.16.0.0/24 is subnetted, 1 subnets D 172.16.1.0 [90/2169856] via 10.1.1.1, 00:00:47, Serial0  
172.16.1.0/24 へのルートがルータ 2 の内部 EIGRP ルートとして示されることに注意してください。

## 集約

EIGRP の集約には 2 つの形式があります。これは 自動集約と手動集約です。

### 自動集約

EIGRP は、2 つの異なるメジャー ネットワークの境界を横断するごとに、自動集約を実行します。たとえば図 13 では、ルータ 2 は 10.0.0.0/8 ネットワークだけをルータ 1 にアドバタイズします。これは、ルータ 2 がルータ 1 に達するために使用するインターフェイスが、別のメジャー ネットワークの中にあるからです。

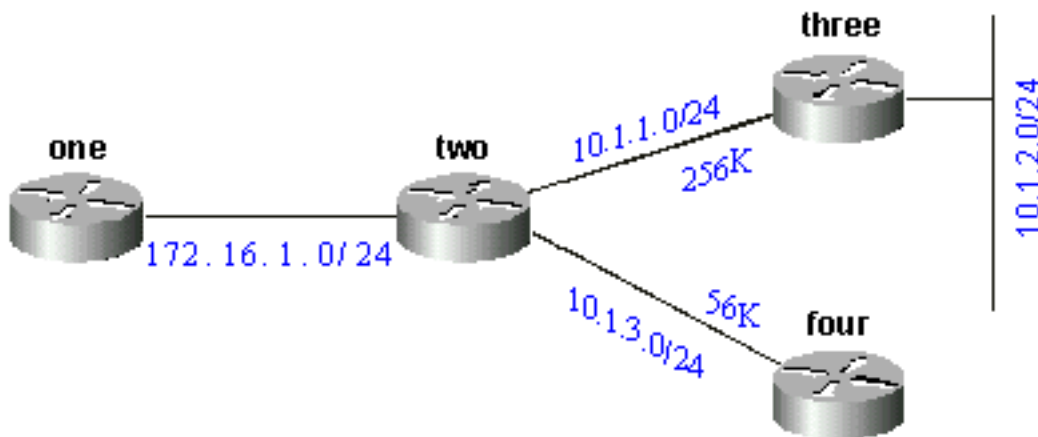


Figure 13

ルータ 1 には次のように表示されます。

```
one# show ip eigrp topology 10.0.0.0 IP-EIGRP topology entry for 10.0.0.0/8 State is Passive,  
Query origin flag is 1, 1 Successor(s), FD is 11023872 Routing Descriptor Blocks: 172.16.1.1  
(Serial0), from 172.16.1.2, Send flag is 0x0 Composite metric is (11023872/10511872), Route is  
Internal Vector metric: Minimum bandwidth is 256 Kbit Total delay is 40000 microseconds  
Reliability is 255/255 Load is 1/255 Minimum MTU is 1500 Hop count is 1
```

このルートはどのような場合も集約ルートとしてはマークされません。これは内部ルートのように見えます。集約ルートの中で、このメトリックは最も優れたメトリックです。10.0.0.0 ネットワークには、帯域幅が 56k のリンクがありますが、このルートの最小帯域幅は 256k であることに注意してください。

集約を実行しているルータでは、集約アドレスに対して null0 へのルートが作成されます。

```
two# show ip route 10.0.0.0 Routing entry for 10.0.0.0/8, 4 known subnets Attached (2  
connections) Variably subnetted with 2 masks Redistributing via eigrp 2000 C 10.1.3.0/24 is  
directly connected, Serial2 D 10.1.2.0/24 [90/10537472] via 10.1.1.2, 00:23:24, Serial1 D  
10.0.0.0/8 is a summary, 00:23:20, Null0 C 10.1.1.0/24 is directly connected, Serial1
```

10.0.0.0/8 へのルートは、Null0 を介して、集約としてマークされます。この集約ルートのトポロ

ジ テーブル エントリは次のようになります。

```
two# show ip eigrp topology 10.0.0.0 IP-EIGRP topology entry for 10.0.0.0/8 State is Passive,
Query origin flag is 1, 1 Successor(s), FD is 10511872 Routing Descriptor Blocks: 0.0.0.0
(Null0), from 0.0.0.0, Send flag is 0x0 (note: the 0.0.0.0 here means this route is originated
by this router) Composite metric is (10511872/0), Route is Internal Vector metric: Minimum
bandwidth is 256 Kbit Total delay is 20000 microseconds Reliability is 255/255 Load is 1/255
Minimum MTU is 1500 Hop count is 0
```

ルータ 2 に、集約の代わりに 10.0.0.0 ネットワークのコンポーネントをアドバタイズさせるには、ルータ 2 の EIGRP プロセスに [no auto-summary](#) を設定します。

## ルータ 2

```
router eigrp 2000 network 172.16.0.0 network 10.0.0.0 no auto-summary
```

[auto-summary](#) をオフにすることで、ルータ 1 には 10.0.0.0 ネットワークのすべてのコンポーネントが表示されます。

```
one# show ip eigrp topology IP-EIGRP Topology Table for process 2000 Codes: P - Passive, A -
Active, U - Update, Q - Query, R - Reply, r - Reply status P 10.1.3.0/24, 1 successors, FD is
46354176 via 20.1.1.1 (46354176/45842176), Serial0 P 10.1.2.0/24, 1 successors, FD is 11049472
via 20.1.1.1 (11049472/10537472), Serial0 P 10.1.1.0/24, 1 successors, FD is 11023872 via
20.1.1.1 (11023872/10511872), Serial0 P 172.16.1.0/24, 1 successors, FD is 2169856 via
Connected, Serial0
```

外部ルートの集約の処理に関して、注意点がいくつかあります。これらについては、後ほど「[外部ルートの自動集約](#)」の項で説明します。

## 手動集約

EIGRP では、手動集約を使用することで、内部ルートと外部ルートを事実上どのビット境界でも集約することが可能です。たとえば図 14 では、ルータ 2 は 192.1.1.0/24、192.1.2.0/24、192.1.3.0/24 を CIDR ブロック 192.1.0.0/22 に集約します。

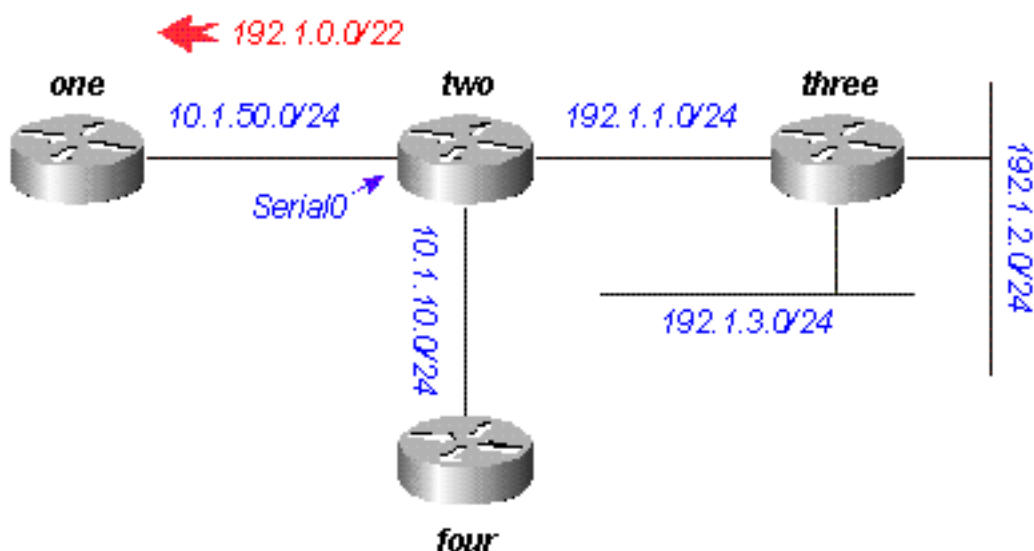


Figure 14

ルータ 2 の設定を次に示します。

```
two# show run .... ! interface Serial0 ip address 10.1.50.1 255.255.255.0 ip summary-address
eigrp 2000 192.1.0.0 255.255.252.0 no ip mroute-cache ! ... two# show ip eigrp topology IP-
EIGRP Topology Table for process 2000 Codes: P - Passive, A - Active, U - Update, Q - Query, R -
Reply, r - Reply status P 10.1.10.0/24, 1 successors, FD is 45842176 via Connected, Loopback0 P
10.1.50.0/24, 1 successors, FD is 2169856 via Connected, Serial0 P 192.1.1.0/24, 1 successors,
FD is 10511872 via Connected, Serial1 P 192.1.0.0/22, 1 successors, FD is 10511872 via Summary
(10511872/0), Null0 P 192.1.3.0/24, 1 successors, FD is 10639872 via 192.1.1.1
(10639872/128256), Serial1 P 192.1.2.0/24, 1 successors, FD is 10537472 via 192.1.1.1
(10537472/281600), Serial1
```

インターフェイス Serial0 下の [ip summary-address コマンド](#)、および Null0 を介した集約ルートに注目してください。ルータ 1 では、以下は内部ルートとして表示されます。

```
one# show ip eigrp topology IP-EIGRP Topology Table for process 2000 Codes: P - Passive, A -
Active, U - Update, Q - Query, R - Reply, r - Reply status P 10.1.10.0/24, 1 successors, FD is
46354176 via 10.1.50.1 (46354176/45842176), Serial0 P 10.1.50.0/24, 1 successors, FD is 2169856
via Connected, Serial0 P 192.1.0.0/22, 1 successors, FD is 11023872 via 10.1.50.1
(11023872/10511872), Serial0
```

## 外部ルートの自動集約

EIGRP は、同じメジャー ネットワークのコンポーネントがあり、それが内部ルートでない限りは、外部ルートの自動集約を実行しません。これについて、図 15 で説明します。

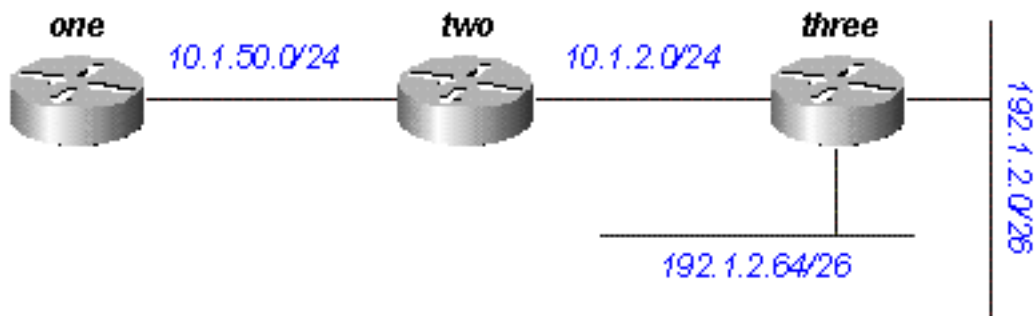


Figure 15

ルータ 3 は、以下の設定に示すとおり、[redistribute connected コマンド](#)を使って外部ルートを 192.1.2.0/26 と 192.1.2.64/26 を EIGRP に挿入します。

### ルータ 3

```
interface Ethernet0 ip address 192.1.2.1 255.255.255.192 ! interface Ethernet1 ip address
192.1.2.65 255.255.255.192!interface Ethernet2 ip address 10.1.2.1 255.255.255.0!router eigrp
2000 redistribute connected network 10.0.0.0 default-metric 10000 1 255 1 1500
```

この設定をルータ 3 に行うと、ルータ 1 のルーティング テーブルの表示は次のようになります。

```
one# show ip route.... 10.0.0.0/8 is subnetted, 2 subnets D 10.1.2.0 [90/11023872] via
10.1.50.2, 00:02:03, Serial0 C 10.1.50.0 is directly connected, Serial0 192.1.2.0/26 is
subnetted, 1 subnets D EX 192.1.2.0 [170/11049472] via 10.1.50.2, 00:00:53, Serial0 D EX
192.1.2.64 [170/11049472] via 10.1.50.2, 00:00:53, Serial0
```

通常は、ルータ 3 が自動集約によって 192.1.2.0/26 および 192.1.2.64/26 ルートを主要なネット送信先 ( 192.1.2.0/24 ) に集約しますが、この場合両方のルートが外部にあるため、ルータ 3 はこれを行いません。しかし、ルータ 2 とルータ 3 間のリンクを 192.1.2.128/26 に再設定して、



このネットワークのネットワーク ステートメントをルータ 2 および 3 に追加した場合は、ルータ 2 上に 192.1.2.0/24 自動集約が生成されます。

### ルータ 3

```
interface Ethernet0 ip address 192.1.2.1 255.255.255.192 ! interface Ethernet1 ip address
192.1.2.65 255.255.255.192!interface Serial0 ip address 192.1.2.130 255.255.255.192 ! router
eigrp 2000 network 192.1.2.0
```

ルータ 2 が 192.1.2.0/24 の集約を生成します。

```
two# show ip route .... D 192.1.2.0/24 is a summary, 00:06:48, Null0....
```

ルータ 1 には集約ルートのみが表示されます。

```
one# show ip route .... 10.0.0.0/8 is subnetted, 1 subnets C 10.1.1.0 is directly connected,
Serial0 D 192.1.2.0/24 [90/11023872] via 10.1.50.2, 00:00:36, Serial0
```

## クエリーの処理と範囲

ルータが近隣ルータからのクエリーを処理するときには、次の規則が適用されます。

クエリー元	ルートの状態	Action
近隣ルータ ( 現行のサクセサ以外 )	パッシブ	現行のサクセサの情報を返信する。
サクセサ	パッシブ	新しいオブジェクトの検索を試行する。成功したら、新しい情報を応答する。失敗したら、宛先を到達不能とマークして、前のサクセサを除くすべての近隣ルータにクエリーを送信する。
任意の近隣ルータ	クエリーより先に、この近隣ルータを通るパスはない。	現時点で最適とされるパスを返信する。
任意の近隣ルータ	クエリー前は不明	宛先が到達不能であると応答する。
近隣ルータ ( 現行のサクセサ以外 )	アクティブ	この宛先に現在サクセサがない場合 ( 通常はそうなります )、到達不能と応答する。
		適切なサクセサがある場合は現在のパス情報を返信する。
サクセサ	アクティブ	新しいオブジェクトの検索を試行する。成功したら、新しい情報を応答する。失敗したら、宛先を到達不能とマークして、前のサクセサを除くすべての近隣ルータにクエリー

を送信する。

上表の各アクションによって、ネットワークが新しいトポロジでコンバートする前に、このクエリーを受信および返信するルータの数が決定されます。そのため、これらのアクションは、ネットワーク内のクエリー範囲に影響します。図 16 の通常の条件下で動作しているネットワークを取り上げて、これらの規則がクエリーの処理方法にどのように影響するかを調べます。

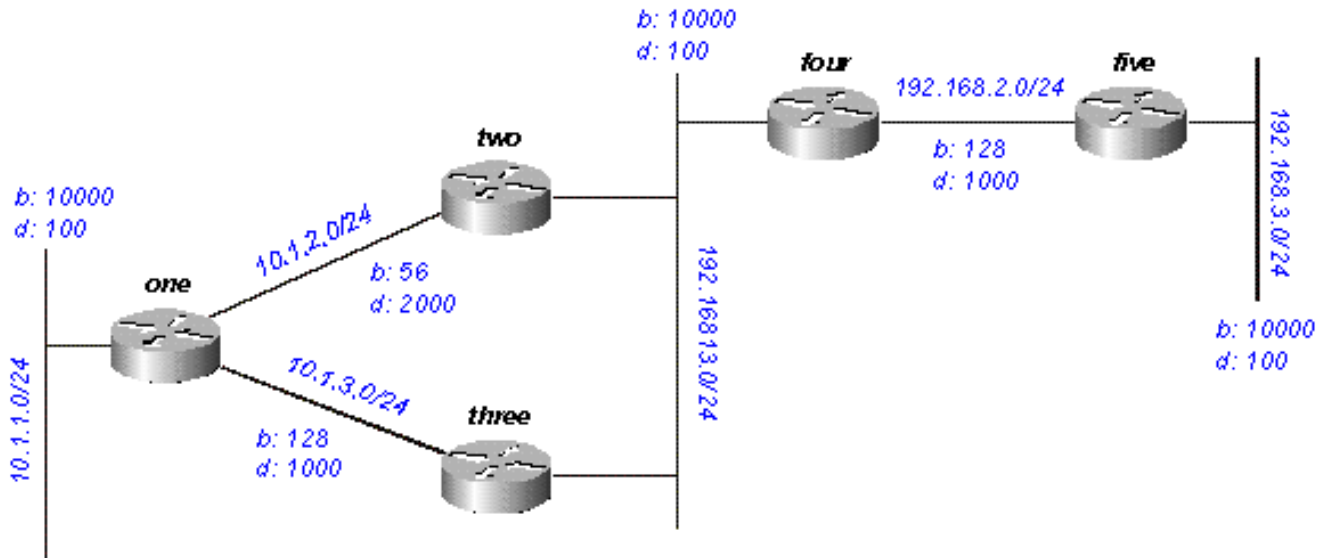


Figure 16

ネットワーク 192.168.3.0/24 (右端) では、次のことが予測されます。

ルータ 1 には、192.168.3.0/24 へのパスが 2 つあります。

ルータ 2 を経由 - 距離 46533485、報告距離 20307200。

ルータ 3 を経由 - 距離 20563200、報告距離 20307200。

ルータ 1 はルータ 3 を通るパスを選択し、ルータ 2 を通るパスをフィージブル サクセサとして保持します。

ルータ 2 および 3 は、ルータ 4 を経由する 192.168.3.0/24 への 1 つのパスを示します。

192.168.3.0/24 に障害が起きたと仮定します。このネットワークにはどのようなアクティビティがみられるのでしょうか。図 16a ~ 16h でこのプロセスを説明します。

ルータ 5 が 192.168.3.0/24 を到達不能にマークし、ルータ 4 に問い合わせます。

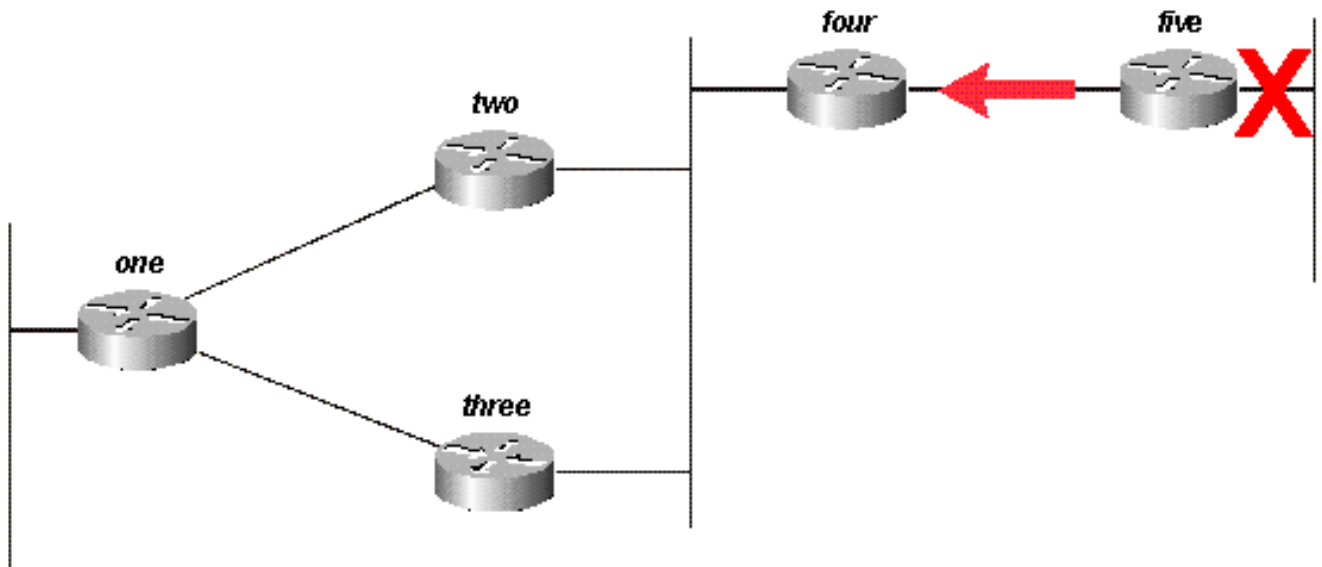


Figure 16a

ルータ 4 は、サクセサからクエリーを受信すると、このネットワークに到達可能な新しいサクセサを探そうとします。見つからなかった場合、ルータ 4 は 192.168.3.0/24 を到達不能にマークし、ルータ 2 と 3 に問い合わせます。

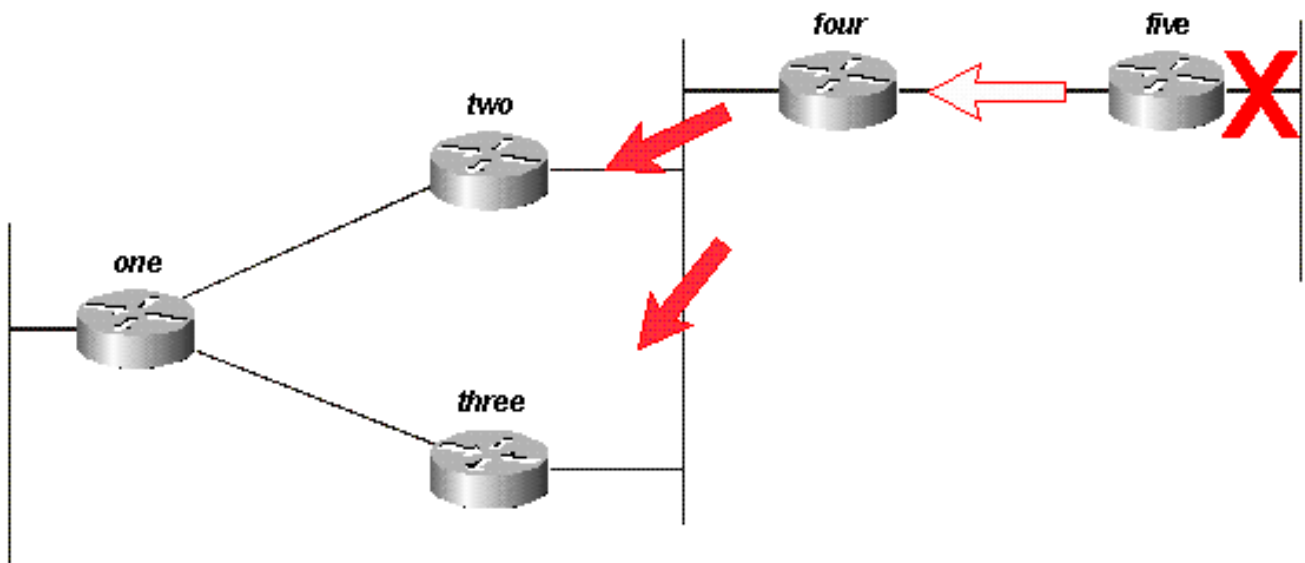


Figure 16b

ルータ 2 と 3 は、192.168.3.0/24 への唯一の到達可能なルートを失ったことを認識し、このルートを到達不能としてマークします。2 と 3 の両方がルータ 1 にクエリーを送信します。

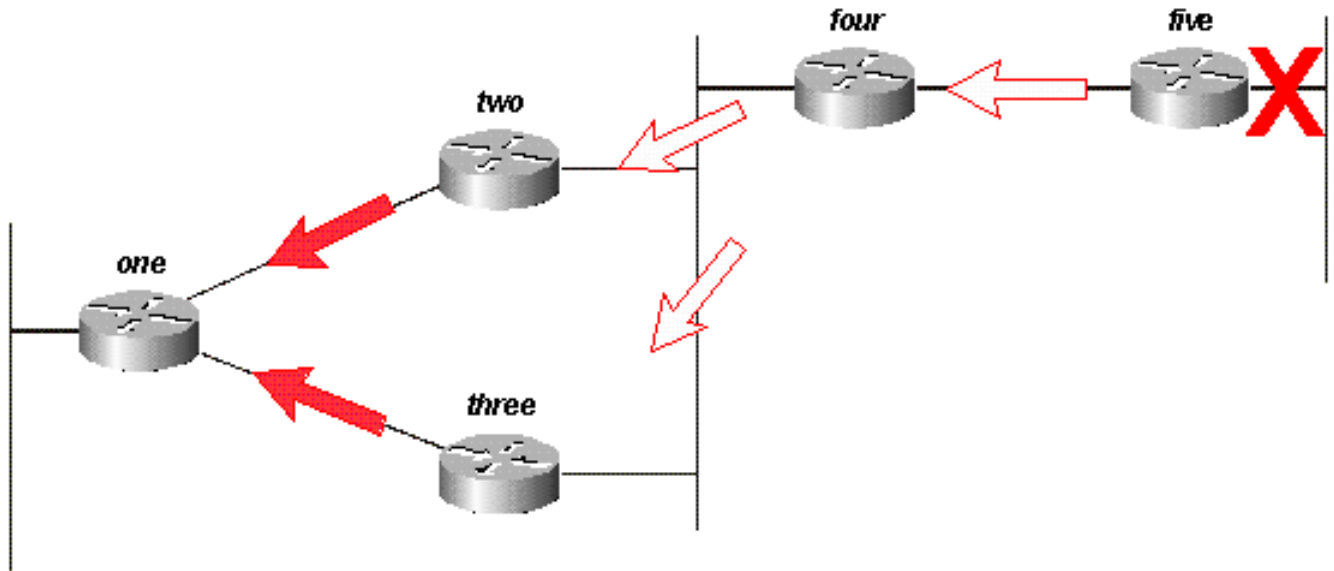


Figure 16c

説明を簡単にするため、ルータ 1 が最初にルータ 3 からクエリーを受信し、このルートを到達不能にマークしたと想定します。するとルータ 1 はルータ 2 からクエリーを受信します。他の順番でも可能ですが、結果は同じになります。

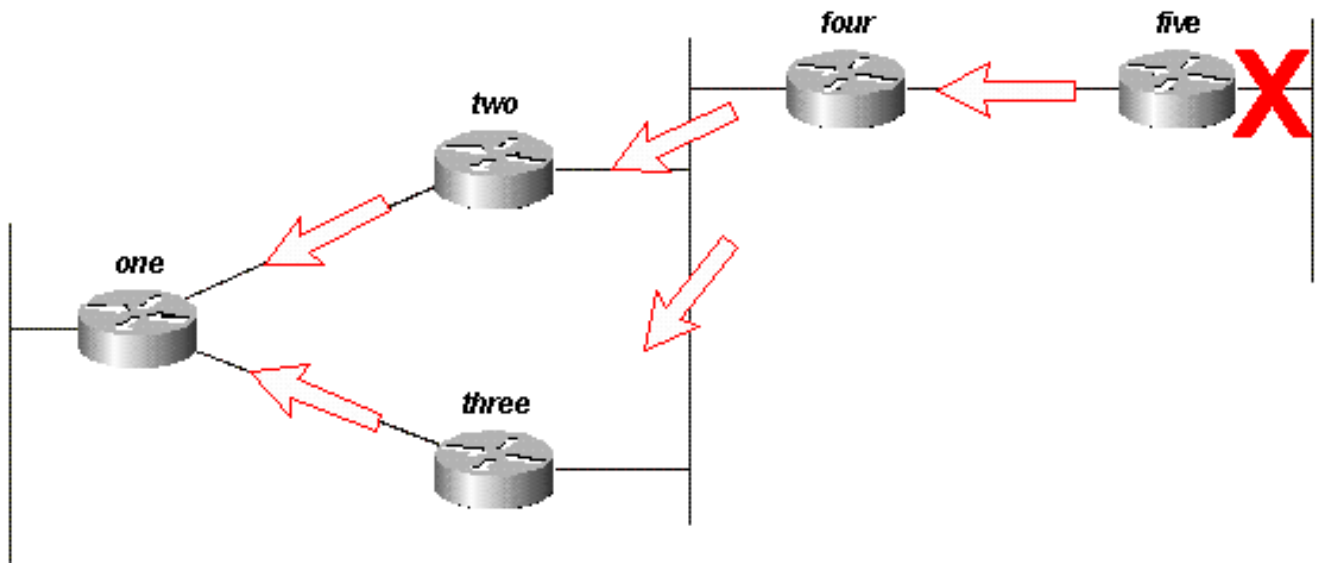


Figure 16d

ルータ 1 は両方のクエリーに対し到達不能として応答します。これでルータ 1 は 192.168.3.0/24 に対してパッシブになります。

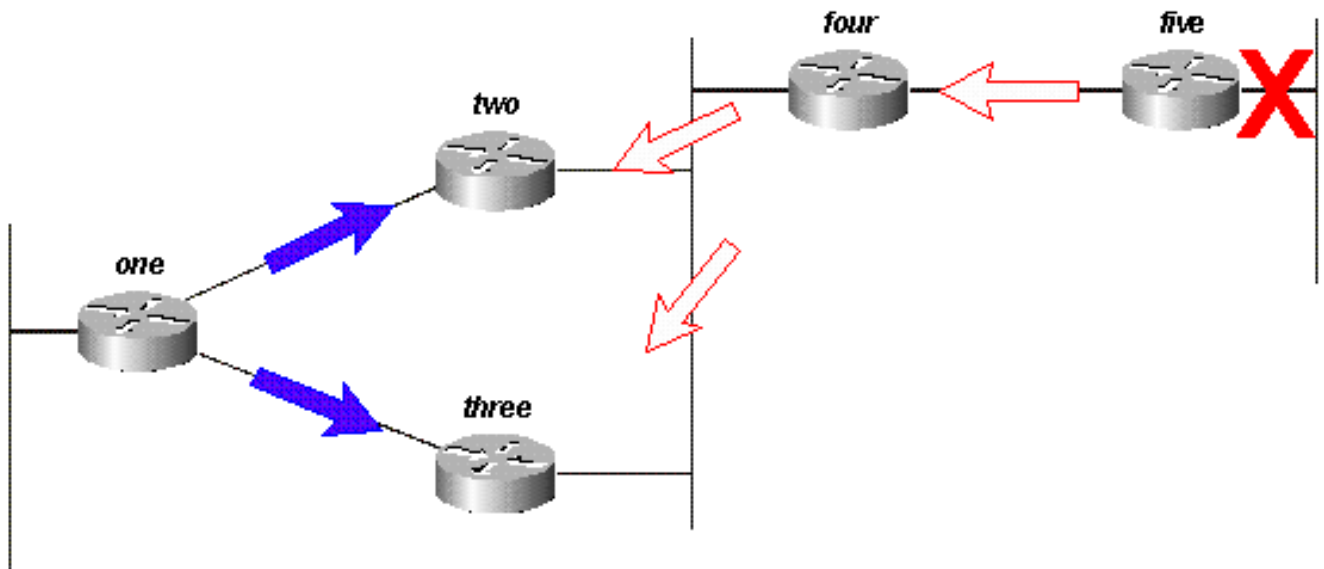


Figure 16e

ルータ 2 と 3 はルータ 4 からのクエリに応答します。このときルータ 2 と 3 は、192.168.3.0/24 に対してパッシブになっています。

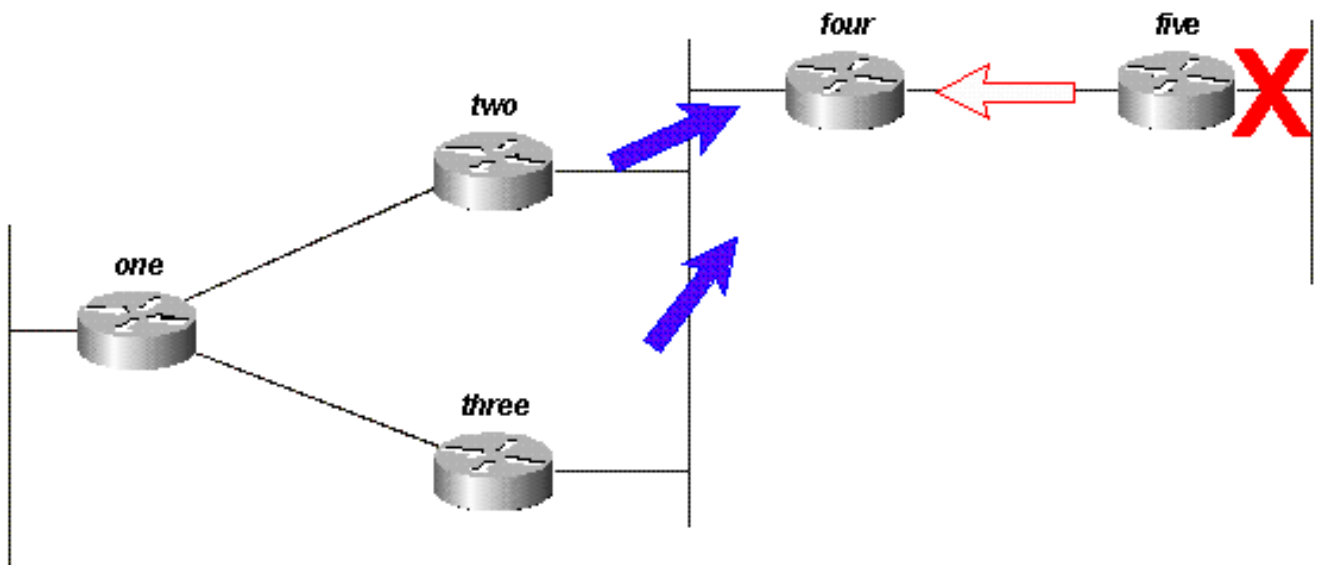


Figure 16f

ルータ 5 は、ルータ 4 からの応答を受信すると、ネットワーク 192.168.3.0/24 をそのルーティングテーブルから削除します。これでルータ 5 は 192.168.3.0/24 に対してパッシブになります。ルータ 5 はルータ 4 にアップデートを送信します。すると、残りのルータのトポロジおよびルーティングテーブルから、このルートが削除されます。

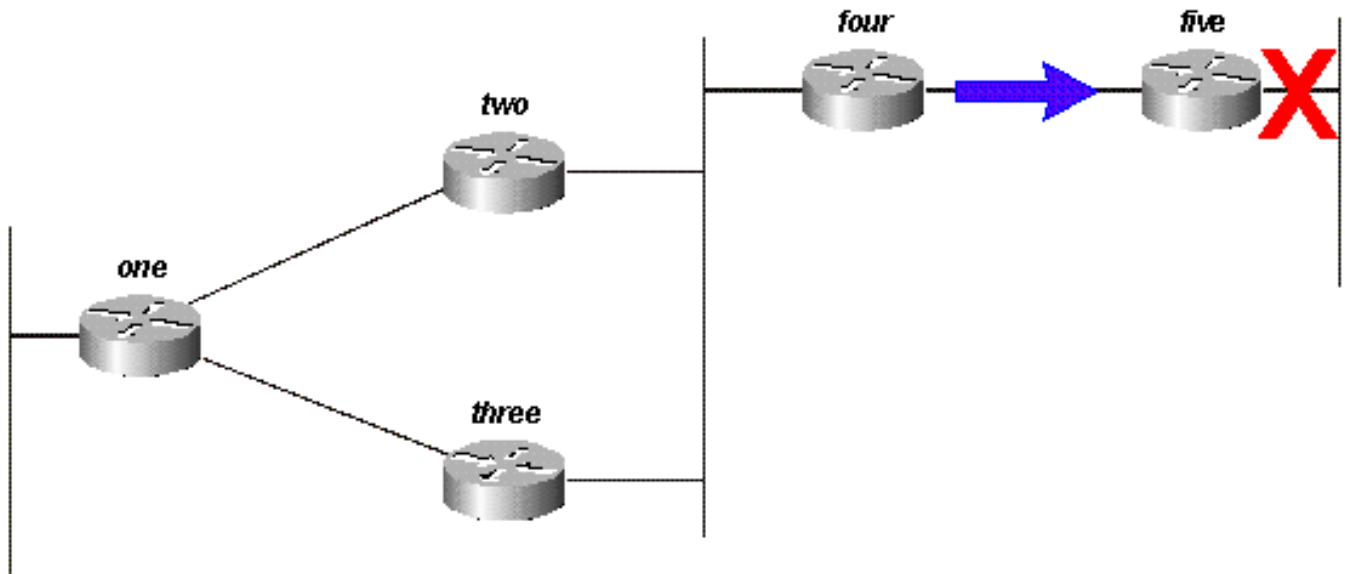


Figure 16g

この他のクエリーパスや処理の順番もあり得ますが、リンクがダウンした場合は、ネットワーク内のすべてのルータがネットワーク 192.168.3.0/24 のクエリーを処理することを理解しておくことが重要です。複数のクエリーを処理することになるルータもあります（この例ではルータ 1）。実際に、クエリーが別の順番でルータに達するのであれば、一部のルータが 3 つまたは 4 つのクエリーを処理することになるはずですが、これは、EIGRP ネットワークでのクエリーの範囲が制限されていないことを示すよい例です。

### 集約ポイントがクエリー範囲に与える影響

次に、同じネットワークにある 10.1.1.0/24 へのパスについて調べます。

ルータ 2 は、10.1.1.0/24 ネットワークへのトポロジ テーブル エントリを保持します。コストはルータ 1 経由で 46251885 です。

ルータ 3 は、10.1.1.0/24 ネットワークへのトポロジ テーブル エントリを保持します。コストはルータ 1 経由で 20281600 です。

ルータ 4 は、ルータ 3 経由で 10.0.0.0/8 ネットワークのトポロジ テーブル エントリを保持し（ルータ 2 と 3 がメジャー ネットワーク境界に自動集約しているため）、メトリックは 20307200 です（ルータ 2 を経由した報告距離は、ルータ 3 を経由した複合メトリックよりも高いため、ルータ 2 経由のパスはフィージブル サクセサではありません）。

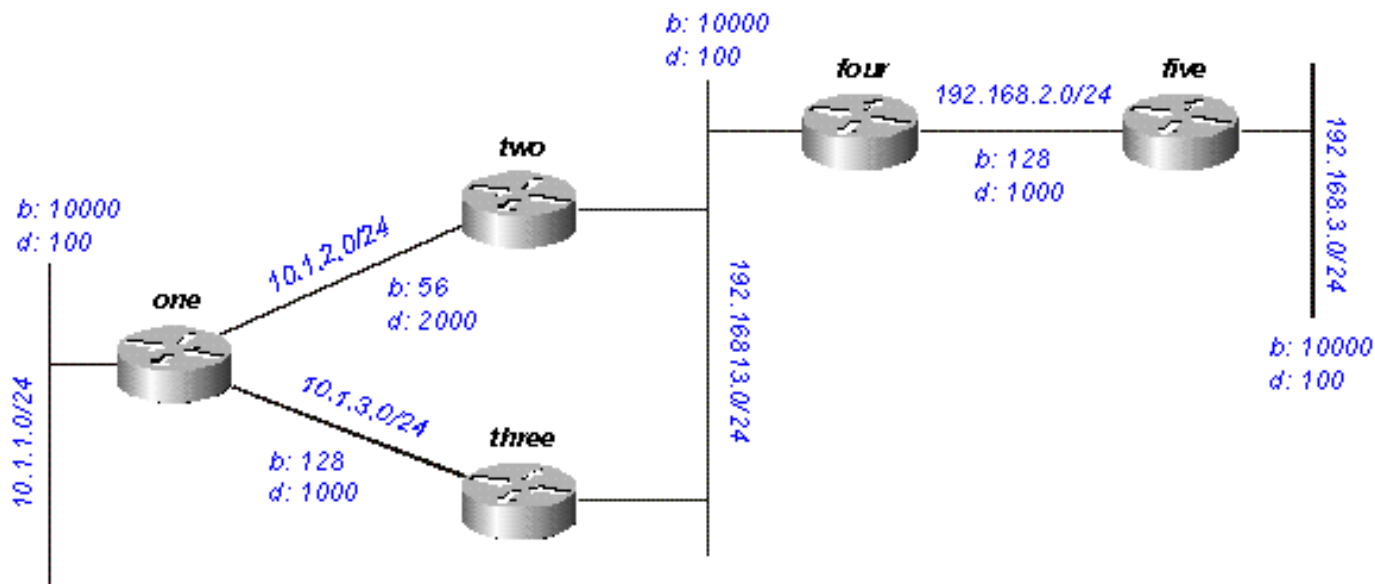


Figure 17

ルータ 1 は  $10.1.1.0/24$  がダウンすると、これを到達不能にマークし、このネットワークへの新しいパスをそれぞれの近隣ルータ (ルータ 2 と 3) に問い合わせます。

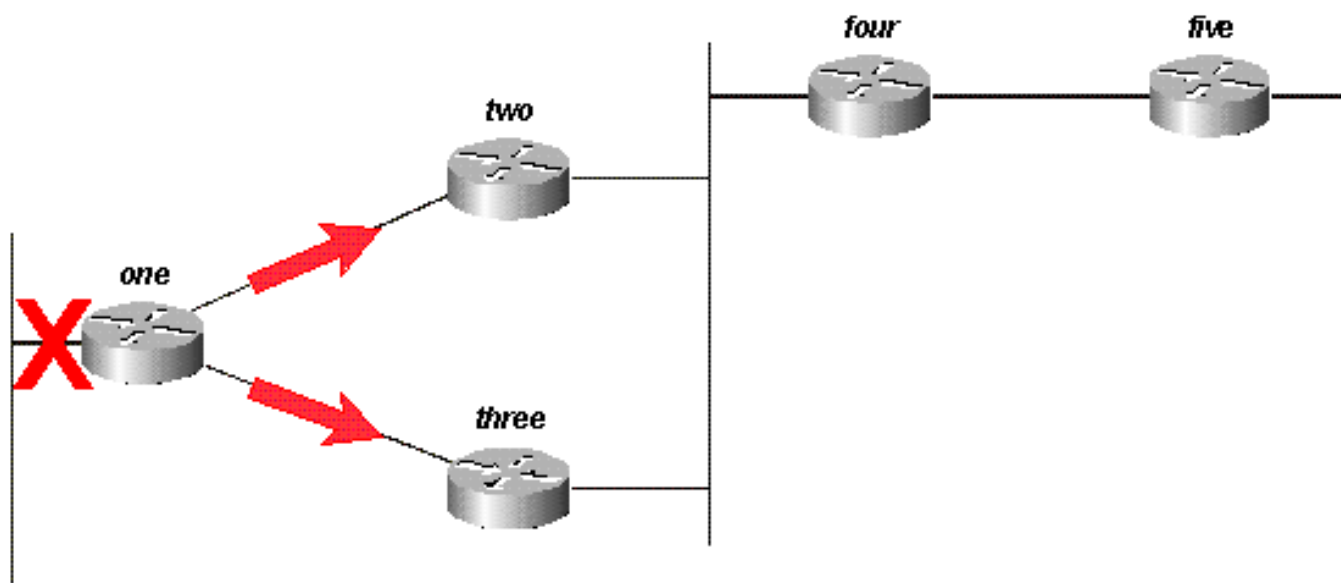


Figure 17a

ルータ 2 は、ルータ 1 からのクエリを受信すると、このルート到達不能にマークし (クエリがルータ 2 のサクセサからのものであるため)、その後ルータ 4 と 3 に問い合わせます。

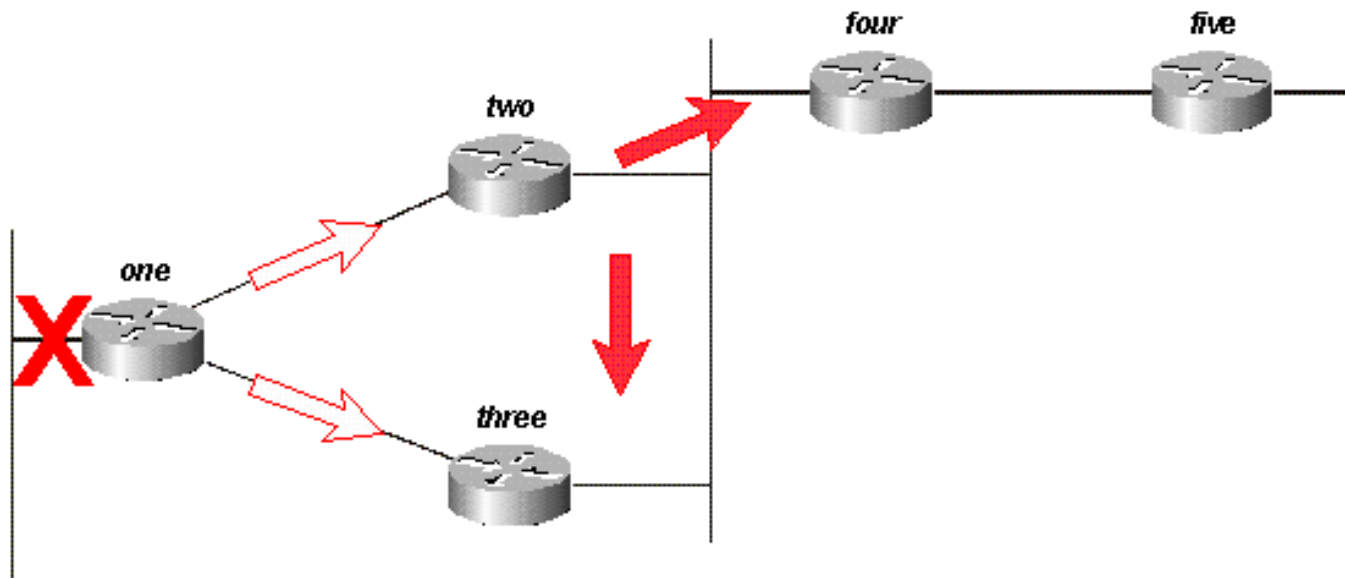


Figure 17b

ルータ 3 は、ルータ 1 からのクエリーを受信すると、この宛先を到達不能にマークし、ルータ 2 と 4 に問い合わせます。

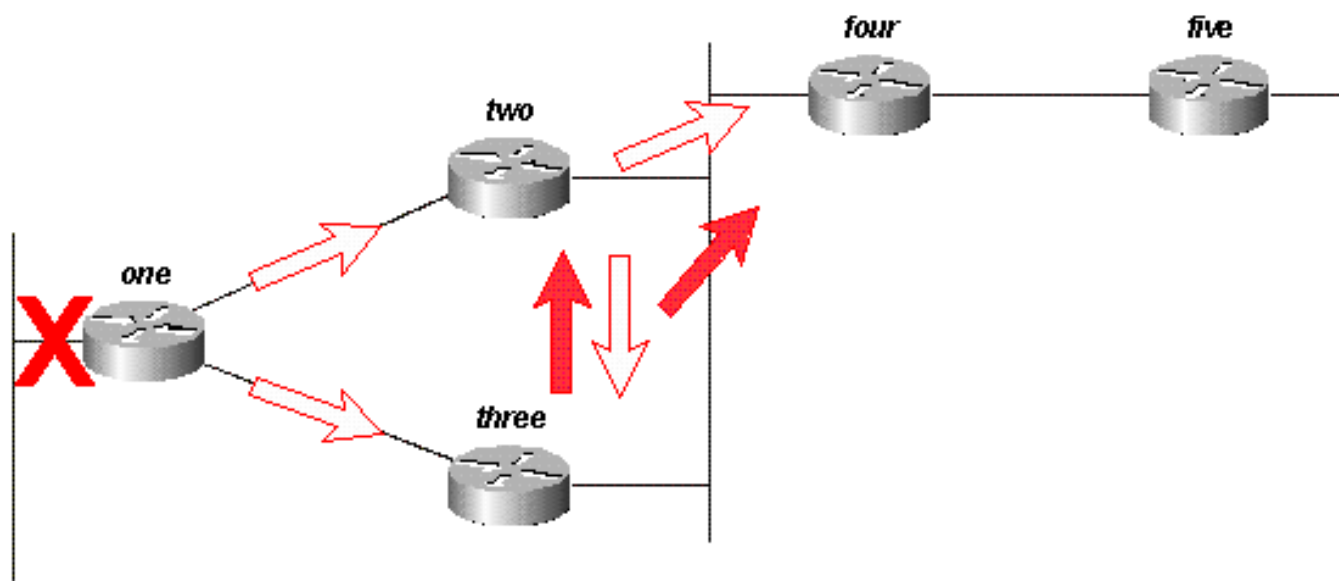


Figure 17c

ルータ 4 は、ルータ 2 および 3 からのクエリーを受信すると、10.1.1.0/24 が到達不能であると応答します。(注：ルータ 4 は 10.0.0.0/8 ルートしか持たないため、問題のサブネットについては知りません。)



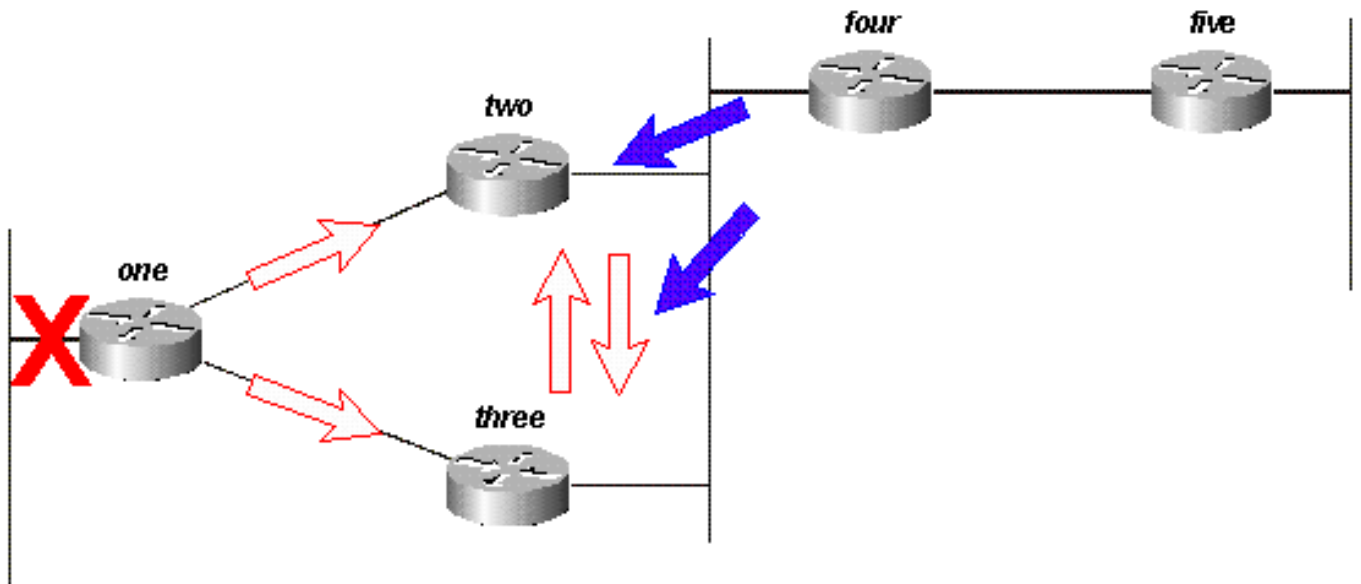


Figure 17d

ルータ 2 と 3 は、10.1.1.0/24 が到達不能であることを相互に応答し合います。

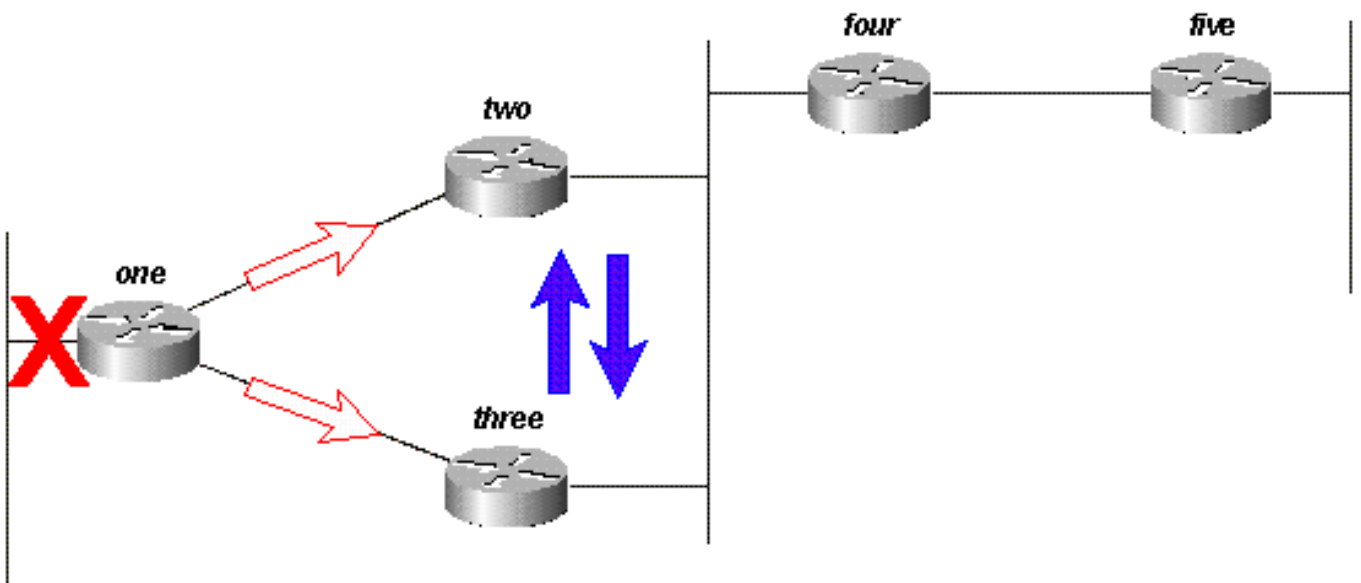


Figure 17e

これでルータ 2 と 3 に未処理のクエリーがなくなったので、両ルータはルータ 1 に 10.1.1.0/24 が到達不能であると応答します。

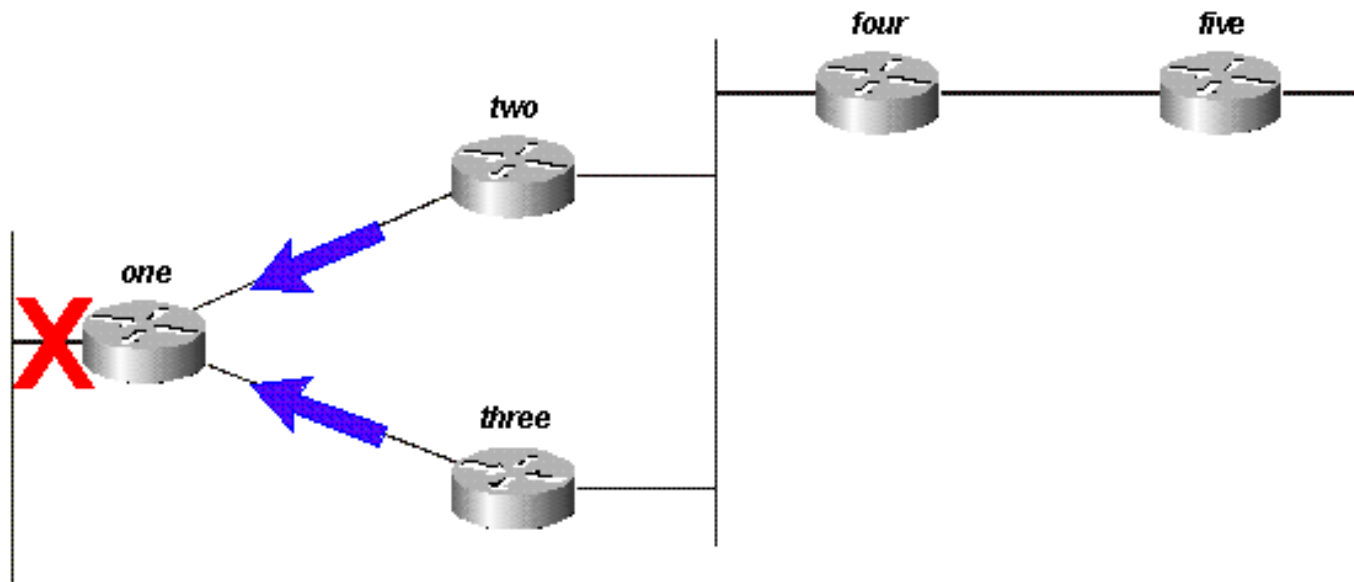


Figure 17f

この場合クエリーは、ルータ 2 と 3 での自動集約によって範囲を制限されています。ルータ 5 はクエリーのプロセスには参加せず、ネットワークの再コンバージェンスにも関与しません。手動集約、自動システム ボーダー、配布リストなどによってクエリーの範囲を制限することも可能です。

### AS の境界がクエリー範囲に与える影響

2 つの EIGRP AS 間でルートを再配送する場合、ルータは通常の処理規則の範囲内でクエリーに回答し、他の AS に新しいクエリーを開始します。たとえば、ルータ 3 に接続するネットワークへのリンクがダウンした場合、ルータ 3 はこのルートを到達不能にマークし、ルータ 2 に対して新しいパスを問い合わせます。

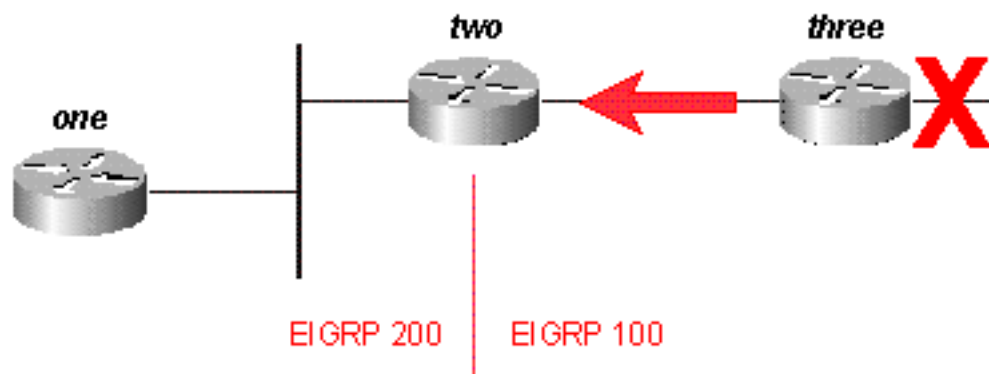


Figure 18a

ルータ 2 はこのネットワークが到達不能であると応答し、AS 200 へのクエリーをルータ 1 に向けはじめます。ルータ 3 は、初めのクエリーに対する応答を受信すると、このルートをテーブルから削除します。ルータ 3 は、このネットワークに対してパッシブになります。

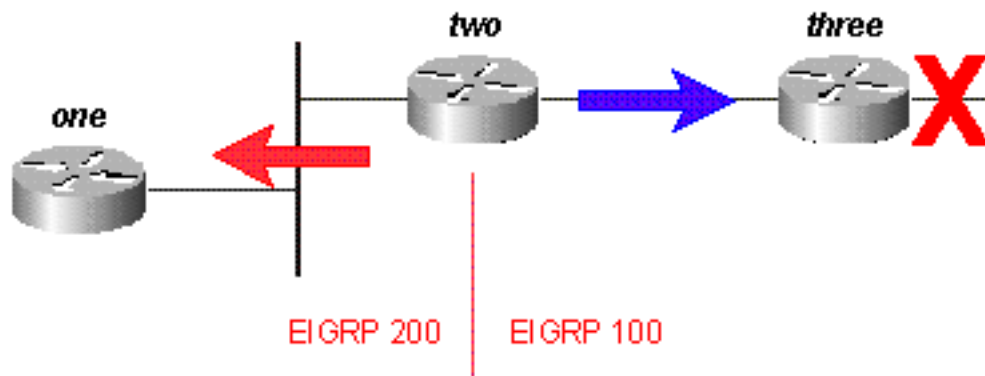


Figure 18b

ルータ 1 がルータ 2 に応答した後、ルートがパッシブになります。

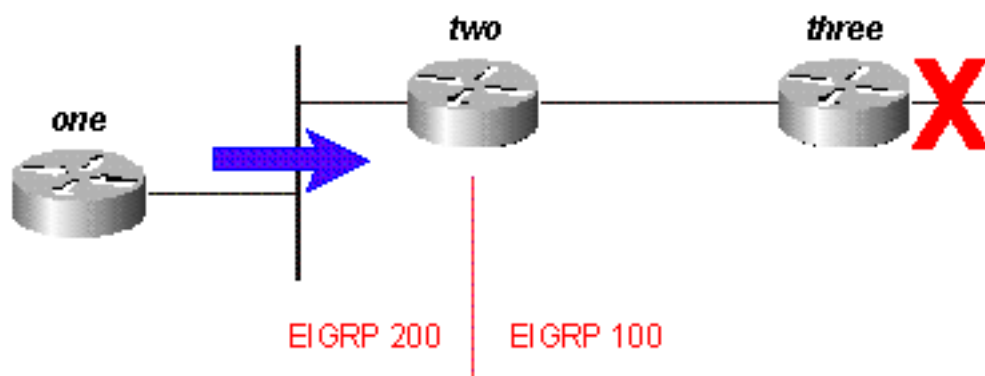


Figure 18c

最初のクエリーが（自律システム境界線によって範囲を制限されていたため）ネットワークの至るところに伝搬されなかったにもかかわらず、そのクエリーは新しいクエリーという形で、2 番目の AS に流出します。この技法では、クエリーに対する応答がなされる前に、このクエリーが通過する必要があるルータ数が制限されるため、ネットワーク内の Stuck In Active ( SIA ) 障害の防止に役立ちます。ただし、それぞれのルータがクエリーを処理しなければならないという全体的な問題が解決するわけではありません。実際、クエリーの範囲を制限するこの手法によって、集約されていたはずのルートの自動集約がなされないために、問題が悪化する場合があります（メジャー ネットワークに外部コンポーネントがなければ、外部ルートは集約されません）。

### 配布リストがクエリー範囲に与える影響

EIGRP の配布リストは、クエリーの伝搬をブロックするのではなく、クエリーに対するどの返信も、到達不能にマークします。図 19 を例にとって説明します。

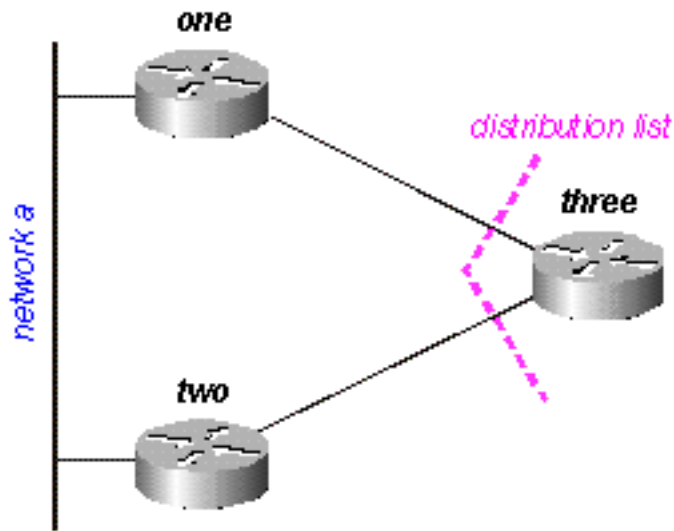


Figure 19

上図では、次のことが言えます。

ルータ 3 には、そのシリアル インターフェイスに対する配布リストが適用されており、これによってネットワーク B をアドバタイズすることだけが許可されています。

ルータ 1 とルータ 2 では、ネットワーク A がルータ 3 経由で到達可能であるとは認識されていません (ルータ 3 はルータ 1 とルータ 2 間の通過ポイントとしては使用されません)。

ルータ 3 は、ルータ 1 をネットワーク A への優先パスとして使用し、ルータ 2 をフェイジブル サクセサとしては扱いません。

ルータ 1 は、ネットワーク A への接続を失うと、このルートを到達不能にマークし、ルータ 3 にクエリーを送信します。ルータ 3 は、そのシリアル ポート上に配布リストがあるため、ネットワーク A へのパスをアドバタイズしません。

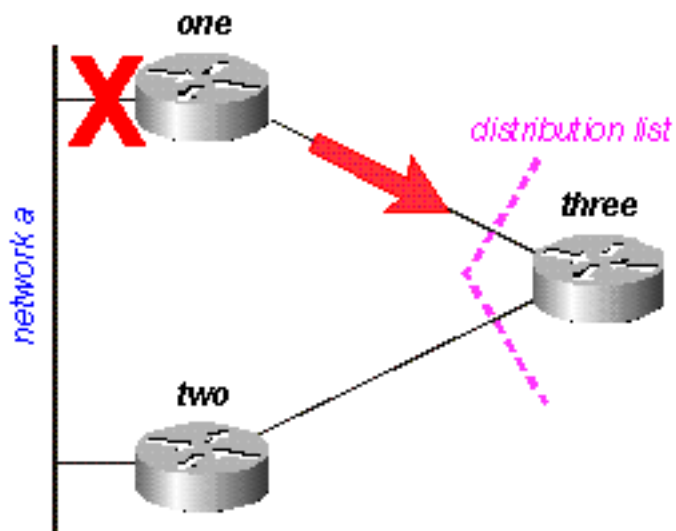


Figure 19a

ルータ 3 はこのルートを送達不能にマークしてから、ルータ 2 に問い合わせます。

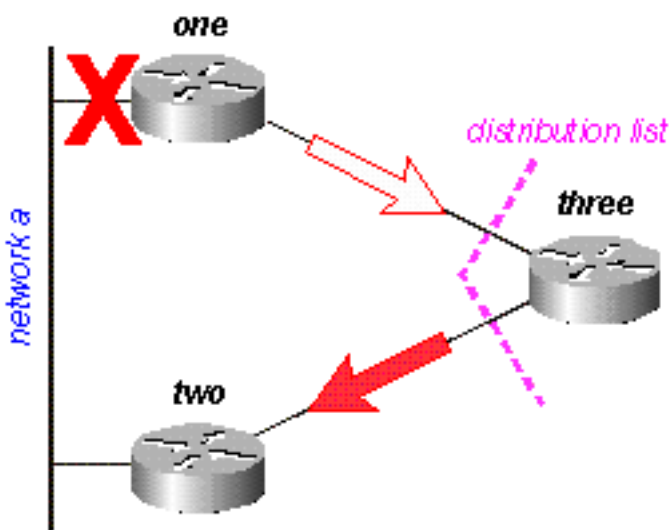


Figure 19b

ルータ 2 はルータ自体のトポロジ テーブルを調べ、ネットワーク A. への有効な接続があることを認識します。このクエリは、ルータ 3 の配布リストによる影響を受けていないことに注意してください。

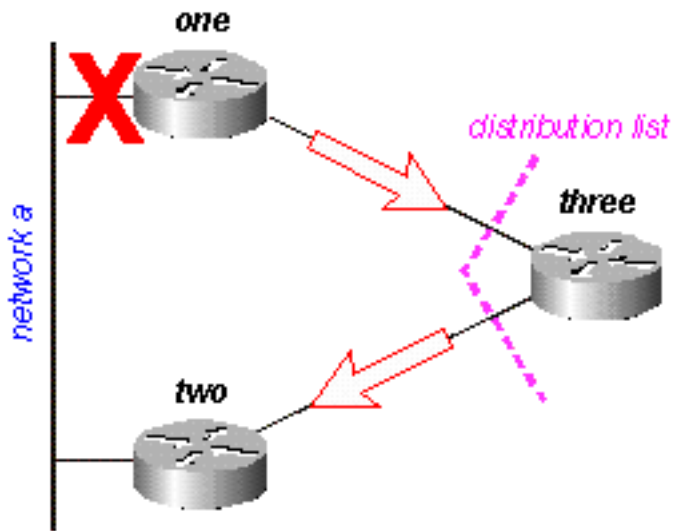


Figure 19c

ルータ 2 は、ネットワーク A は到達不能であると応答します。ルータ 3 には有効なルートがあります。

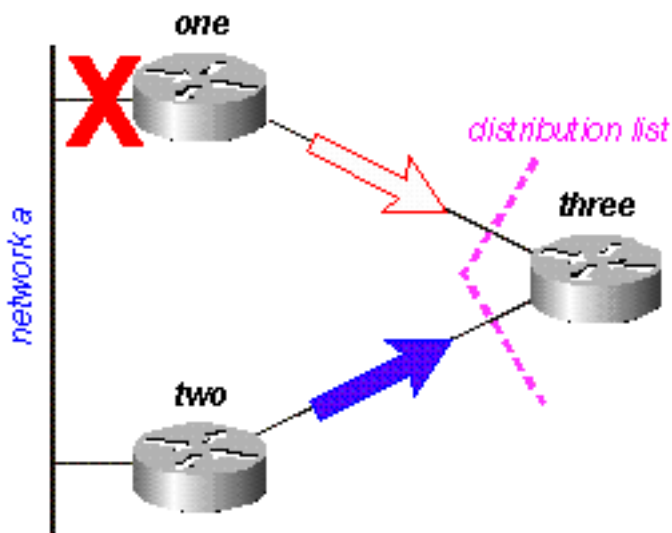


Figure 19d

ルータ 3 は、ルータ 1 からのクエリーに対する応答を作成します。しかし、配布リストがあるために、ルータ 3 は、実際にはネットワーク A への有効なルートがあるにもかかわらず、ネットワーク A が到達不能と応答してしまいます。

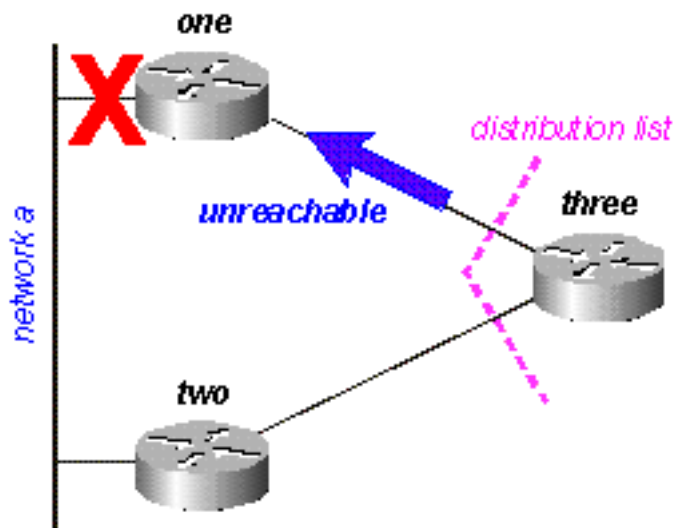


Figure 19e

## パケットのペーシング

一部のルーティングプロトコルは、コンバージ中に（ネットワークの変更に適合している間）、低帯域幅リンクで使用可能なすべての帯域幅を消費します。EIGRPはネットワーク上でのパケットの送信速度をペーシングし、使用可能帯域幅の一部のみを使用するようにすることで、この輻輳を回避します。EIGRPのデフォルト設定は、使用可能帯域幅の最大50%までの使用となっています。しかしこれは、次のコマンドを使って変更できます。

```
router(config-if)# ip bandwidth-percent eigrp 2 ? <1-999999> Maximum bandwidth percentage that EIGRP may use
```

EIGRPは、インターフェイス上に送信するパケットをキューに入れる場合は基本的に毎回、次の公式を使用して、パケットを送信するまでの待ち時間を割り出します。

$$(8 * 100 * \text{パケット サイズ [バイト]}) / (\text{帯域幅 [kbps]} * \text{帯域幅 [\%]})$$

たとえば、帯域幅 56k のシリアル インターフェイスで送信する 512 バイトのパケットをキューに入れる場合、EIGRP は待機します。

$$(8 * 100 * 512 \text{ バイト}) / (56000 \text{ bps} * \text{帯域幅 } 50\%) = (8 * 100 * 512) / (56000 * 50) = 409600 / 2800000 = 0.1463 \text{ 秒}$$

これにより、EIGRP がそのパケットを送信する前に、少なくとも 512 バイトの 1 パケット（またはパケットのグループ）をこのリンクで送信することができます。パケットの送信されるタイミングはペーシング タイマーが決定し、通常ミリ秒単位でこれを表示します。上記の例では、パケットのペーシング タイムは 0.1463 秒です。 [show ip eigrp interface](#) には、次に示すとおり、ペーシング タイマーを表示するフィールドがあります。

```
router# show ip eigrp interface IP-EIGRP interfaces for process 2 Xmit Queue Mean Pacing Time
Multicast Pending Interface Peers Un/Reliable SRTT Un/Reliable Flow Timer Routes
Se0 1 0/0 28 0/15
127 0Se1 1 0/0 44 0/15 211 0router#
```

表示されるタイムは、このインターフェイスで送信可能な最大のパケットを指す、maximum

transmission unit ( MTU; 最大伝送ユニット ) のペーシング間隔です。

## デフォルト ルーティング

EIGRP へデフォルト ルートを挿入するには 2 つの方法があります。スタティック ルートを再配布するか、または 0.0.0.0/0 に集約します。未知の宛先へのすべてのトラフィックを、ネットワーク コアのデフォルト ルートに向けたい場合は、前者の方法を使用してください。この方法は、インターネットへの接続をアダプタイズする場合に効果的です。次に、例を示します。

```
ip route 0.0.0.0 0.0.0.0 x.x.x.x (next hop to the internet) ! router eigrp 100 redistribute
static default-metric 10000 1 255 1 1500
```

EIGRP に再配布されるスタティック ルートは、ネットワーク 0.0.0.0 である必要はありません。別のネットワークを使用する場合は、[ip default-network コマンド](#)を使用して、このネットワークをデフォルト ネットワークとしてマークする必要があります。詳細は『[最後の手段としてのゲートウェイの設定](#)』を参照してください。

デフォルト ルートへの集約は、リモート サイトにデフォルト ルートを供給する場合にのみ有効です。集約はインターフェイスごとに設定されるので、配布リストやその他のメカニズムを使用して、デフォルト ルートのネットワーク コアへの伝搬を防ぐ必要はありません。0.0.0.0/0 への集約は、他のルーティング プロトコルから学習したデフォルト ルートを無効にすることに注意してください。この方式でルータにデフォルト ルートを設定する唯一の方法は、スタティック ルートを 0.0.0.0/0 に設定することです。( Cisco IOS ソフトウェア 12.0(4)T 以降では、[ip summary-address eigrp コマンド](#)の末尾でアドミニストレーティブ ディスタンスも設定できるようになったため、0.0.0.0/0 ルートがローカル 集約によってオーバーライドされることはありません。)

```
router eigrp 100 network 10.0.0.0 ! interface serial 0 encapsulation frame-relay no ip
address ! interface serial 0.1 point-to-point ip address 10.1.1.1 frame-relay interface-dlci
10 ip summary-address eigrp 100 0.0.0.0 0.0.0.0
```

## ロード バランシング

EIGRP がルーティング テーブルに最大 4 つまでの等コスト ルートを挙げた時点で、これらのルートはルータによってロード バランスされます。ロード バランシングのタイプ ( パケット単位または宛先単位 ) は、ルータで実行されるスイッチングのタイプによります。しかし、EIGRP は不等コストのリンクでもロード バランスを行います。

**注:** `max-paths` を使用すれば、EIGRP に最大 6 つまでの等コストのルートの使用を設定できます。

特定の宛先へのパスが 4 つあり、これらのパスのメトリックが次のとおりであるとします。

path 1: 1100

path 2: 1100

path 3: 2000

path 4: 4000



ルータはデフォルトでパス 1 とパス 2 の両方にトラフィックを流します。また、EIGRP を使用することで、[variance コマンド](#)を使用して、ルータにトラフィックをパス 3 とパス 4 にも流すように指示することができます。バリエーションは乗数です。トラフィックは、バリエーションによって乗算される最適パスよりも小さいメトリックを持つ、任意のリンクに流されます。パス 1、2、3 のロード バランスを行うため、バリエーション 2 を使用します。これは、 $1100 \times 2 = 2200$  でパス 3 を通過するメトリックよりも大きくなるためです。同様に、パス 4 も追加するには、`router eigrp` コマンドでバリエーション 4 を実行してください。『[IGRP および EIGRP における不等コストパスの負荷バランシング \(バリエーション\)](#)』を参照してください。参照してください。

トラフィックは、これらのルータのパス間でどのように分配されるのでしょうか。ルータは、各パスを通過するメトリックで最大メトリックを割り、その結果を最も近い整数値に切り捨て、この数値をトラフィックシェア カウントとして使用します。

```
router# show ip route 10.1.4.0 Routing entry for 10.1.4.0/24 Known via "igrp 100", distance 100, metric 12001 Redistributing via igrp 100, eigrp 100 Advertised by igrp 100 (self originated) eigrp 100 Last update from 10.1.2.2 on Serial1, 00:00:42 ago Routing Descriptor Blocks: * 10.1.2.2, from 10.1.2.2, 00:00:42 ago, via Serial1 Route metric is 12001, traffic share count is 1 Total delay is 20010 microseconds, minimum bandwidth is 1000 Kbit Reliability 1/255, minimum MTU 1 bytes Loading 1/255, Hops 0
```

次の例では、トラフィックシェア カウントは次のようになります。

パス 1 とパス 2 :  $4000/1100 = 3$

パス 3 :  $4000/2000 = 2$

パス 4 :  $4000/4000 = 1$

ルータは最初の 3 つのパケットをパス 1 経由で送信し、次の 3 つのパケットをパス 2 経由で送信し、その次の 2 つのパケットをパス 3 経由で送信し、次の 1 つのパケットをパス 4 経由で送信します。それから再び、次の 3 パケットをパス 1 から送信し、これを繰り返します。

注: EIGRP はバリエーションを設定している場合でも、当該ルートの到達可能距離よりも報告距離の方が大きい場合は、不等コストではトラフィックを送信しません。詳細は、「[到達可能距離、報告距離、およびフィジブル サクセス](#)」の項を参照してください。

## [メトリックの使用](#)

EIGRP の初期設定で EIGRP メトリックを調整する場合は、次の 2 つの基本的な規則を覚えておいてください。

帯域幅は、常にインターフェイスの実際の帯域幅に設定する必要があります。ただし、マルチポイントシリアルリンク、またはその他のメディア速度のミスマッチ状況は、この規則の例外となります。

EIGRP のルーティング決定を調整する場合は、常に遅延を使用する必要があります。

EIGRP は、インターフェイス帯域幅を使用してパケットの送信速度の決定するので、これらの設

定を正しく行うことが重要です。EIGRP の選択するパスに影響を与える必要がある場合は、常に遅延を使用してください。

帯域幅が小さいほど、帯域幅が複合メトリックに与える影響は高くなります。帯域幅が大きいほど、遅延が複合メトリックに与える影響は高くなります。

## 再配布における管理タグの使用

外部管理タグは、EIGRP とその他のプロトコル間の再配送ルーティング ループを切断するのに役立ちます。ルートが EIGRP に再配送される時、このルートにタグングすることで、EIGRP から外部プロトコルへの再配送をブロックできます。EIGRP では、外部ルートから学習されたデフォルト ゲートウェイのアドミストレーティブ ディスタンスを変更することはできません。これは、EIGRP ではアドミストレーティブ ディスタンスの変更は内部ルートだけに適用されるためです。メトリックを増加するには、route-map と prefix-list を使用します。アドミストレーティブ ディスタンスは変更しないでください。次に、これらのタグの基本的な設定例を示します。ただし、この例は再配布ループの切断に使用されるすべての設定を示すわけではありません。

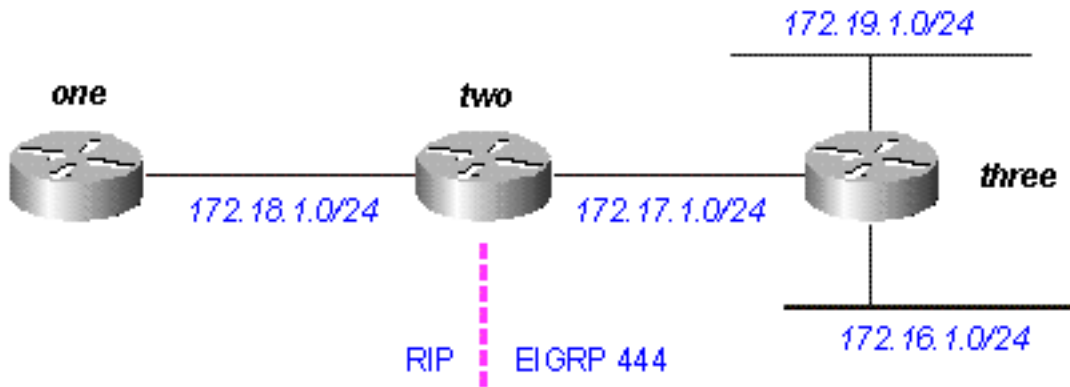


Figure 20

EIGRP に接続するルートを再配送するルータ 3 には、次のように表示されます。

```
three# show run .... interface Loopback0 ip address 172.19.1.1 255.255.255.0 ! interface
Ethernet0 ip address 172.16.1.1 255.255.255.0 loopback no keepalive ! interface Serial0 ip
address 172.17.1.1 255.255.255.0 .... router eigrp 444 redistribute connected route-map foo
network 172.17.0.0 default-metric 10000 1 255 1 1500 .... access-list 10 permit 172.19.0.0
0.0.255.255 route-map foo permit 10 match ip address 10 set tag 1 .... three# show ip eigrp topo
IP-EIGRP Topology Table for process 444 Codes: P - Passive, A - Active, U - Update, Q - Query, R
- Reply, r - Reply status P 172.17.1.0/24, 1 successors, FD is 2169856 via Connected, Serial0
via Redistributed (2169856/0) P 172.16.1.0/24, 1 successors, FD is 281600 via Redistributed
(281600/0) P 172.19.1.0/24, 1 successors, FD is 128256, tag is 1 via Redistributed (128256/0)
```

EIGRP からのルートを RIP に再配布するルータ 2 には、次のように表示されます。

```
two# show run .... interface Serial0 ip address 172.17.1.2 255.255.255.0 ! interface Serial1 ip
address 172.18.1.3 255.255.255.0 .... router eigrp 444 network 172.17.0.0 ! router rip
redistribute eigrp 444 route-map foo network 10.0.0.0 network 172.18.0.0 default-metric 1 ! no
ip classless ip route 1.1.1.1 255.255.255.255 Serial0 route-map foo deny 10 match tag 1 ! route-
map foo permit 20 .... two# show ip eigrp topo IP-EIGRP Topology Table for process 444 Codes: P
- Passive, A - Active, U - Update, Q - Query, R - Reply, r - Reply status P 172.17.1.0/24, 1
successors, FD is 2169856 via Connected, Serial0 P 172.16.1.0/24, 1 successors, FD is 2195456
```

```
via 172.17.1.1 (2195456/281600), Serial0 P 172.19.1.0/24, 1 successors, FD is 2297856, tag is 1
via 172.17.1.1 (2297856/128256), Serial0
```

172.19.1.0/24 のタグ 1 に注目してください。

ルータ 2 によって再配送される RIP ルートを受信するルータ 1 には、次のように表示されます。

```
one# show run.... interface Serial0 ip address 172.18.1.2 255.255.255.0 no fair-queue clockrate
1000000 router rip network 172.18.0.0 .... one# show ip route Gateway of last resort is not set
R 172.16.0.0/16 [120/1] via 172.18.1.3, 00:00:15, Serial0 R 172.17.0.0/16 [120/1] via
172.18.1.3, 00:00:15, Serial0 172.18.0.0/24 is subnetted, 1 subnets C 172.18.1.0 is directly
connected, Serial0
```

172.19.1.0/24 が存在しないことに注意してください。

## EIGRP コマンド出力について

### show ip eigrp traffic

このコマンドは、EIGRP 名前付き設定と EIGRP 自律システム (AS) の設定に関する情報を表示するために使用されます。このコマンドの出力には、隣接 EIGRP ルータ間で交換された情報が示されます。次の表の下に、各出力フィールドの説明を示します。

<u>show ip eigrp traffic</u>
EIGRP-IPv4 Traffic Statistics for AS (11)
Hellos sent/received: 1927/1930
Updates sent/received: 20/39
Queries sent/received: 10/18
Replies sent/received: 18/16
Acks sent/received: 66/41
SIA-Queries sent/received: 0/0
SIA-Replies sent/received: 0/0
Hello Process ID: 270
PDM Process ID: 251
Socket Queue: 0/2000/1/0 (current/max/highest/drops)
Input Queue: 0/2000/1/0 (current/max/highest/drops)

### 設定に関する説明

Hellos sent/received は、送受信された hello パケットの数を示します ( sent -1927/received -1930 )。

Updates sent/received は、送受信された更新の数を示します ( sent-20/received-39 )。

Queries sent/received は、送受信されたクエリー パケットの数を示します ( sent-10/received-18 )。

Replies sent/received は、送受信された応答パケットの数を示します ( sent-18/received-16 )  
。

Acks sent/received は、送受信された確認応答パケットの数を示します ( sent-66/received-41 )。

SIA-Queries sent/received は、送受信された stuck in active クエリー パケットの数を示します ( sent-0/received-0 )。

SIA-Replies sent/received は、送受信された stuck in active 応答パケットの数を示します ( sent-0/received-0 )。

Hello Process ID は、hello プロセスの ID です ( 270 )。

PDM Process ID は、プロトコル依存型モジュール IOS プロセスの ID を示します ( 251 )。

Socket Queue は、IP から EIGRP Hello Process へのソケット キュー カウンタを示します ( current-0/max-2000/highest-1/drops-0 )。

Input Queue は、EIGRP Hello Process から EIGRP PDM へのソケット キュー カウンタを示します ( current-0/max-2000/highest-1/drops-0 )。

## [show ip eigrp topology](#)

このコマンドはフィージブル サクセサのみを表示します。トポロジ テーブルのすべてのエントリを表示するには、[show ip eigrp topology all-links コマンド](#)を使用します。次の表の下に、各出力フィールドの説明を示します。

```
show ip eigrp topology
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status

A 10.2.4.0/24, 0 successors, FD is 512640000, tag is 0x0, Q
   1 replies, active 00:00:01, query-origin: Local origin
     via 10.1.2.2 (Infinity/Infinity), r, Q, Serial1

   Remaining replies:
     via 10.1.1.2, r, Serial0

P 10.3.9.0/24, 1 successors, FD is 512640000, U
  * via 10.1.2.2 (512640000/128256), Serial1
```

## 設定に関する説明

A は「アクティブ」を意味します。ここに「P」が表示されていれば、「パッシブ」を意味します。

10.2.4.0/24 は宛先またはマスクです。

**0 successors** は、この宛先に使用可能なサクセサ (またはパス) の数を示します。  
**Successors** が大文字の場合、このルートが移行中であることを示します。

**FD is 512640000** は、到達可能距離を示します。到達可能距離は、この宛先に到達するための最適なメトリック、もしくはルータがアクティブになった時点での既知の最適なメトリックです。

**tag is 0x0** は、**set tag** および **match tag** コマンド でルートマップを使用して設定またはフィルタできます。

**Q** は、クエリーが保留中であることを意味します。このフィールドには、**U** (更新保留中) または **R** (応答保留中) が表示されることもあります。

**1 replies** は、未処理の応答数を示します。

**active 00:00:01** は、このルートがアクティブであった期間を指しています。

**query origin: Local origin** は、クエリーを発信したルートを示します。このフィールドには、**Multiple origins** (サクセサではなく複数の近隣ルータがこの宛先に対してクエリーを送信した) または **Successor origin** (サクセサがクエリーを発信した) が表示されることもあります。

**via 10.1.2.2** は、IP アドレス 10.1.2.2 の近隣ルータからこのルートを学習したことを示します。このフィールドには、**Connected** (ネットワークがこのルータに直接接続している場合)、**Redistributed** (このルートがこのルータの EIGRP に再配布される場合)、または **Summary** (このルートがこのルータで生成された集約ルータである場合) が表示されることもあります。

**((Infinity/Infinity)** は、最初のフィールドでこの近隣ルータを介してこのパスに到達するためのメトリックを示し、および 2 番目のフィールドでこの近隣ルータを介して報告された距離を示します。

**r** は、このネイバー ルータへのクエリーをすでに完了し、応答を待っていることを示します。

**Q** はこのルートの送信フラグで、保留中のクエリーがあることを意味します。このフィールドには、**U** は、保留中の更新があることを意味します。**R** は、保留中の応答があることを意味します。

**Serial1** は、この近隣ルータに到達できるインターフェイスです。

**Via 10.1.1.2** は、応答の待たれる近隣ルータを示します。

**r** は、この近隣ルータに対してルートを問い合わせたが、まだ応答を受信していないことを示

します。

Serial0 は、この近隣ルータに到達できるインターフェイスです。

Via 10.1.2.2(512640000/128256), Serial1 は、使用するはこのルートであることを示しています ( 次のパスまたは宛先が、複数の等コストのルートのうち、どのパスを通るかを示します。 )

## [show ip eigrp topology <network>](#)

このコマンドは、フィージブル サクセサだけでなく、この宛先のトポロジ テーブルにあるすべてのエントリを表示します。 次の表の下に、各出力フィールドの説明を示します。

```
show ip eigrp topology network
IP-EIGRP topology entry for 20.0.0.0/8
State is Passive, Query origin flag is 1, 2 Successor(s), FD is 307200
Routing Descriptor Blocks:
10.1.1.2 (Ethernet1), from 10.1.1.2, Send flag is 0x0
  Composite metric is (307200/281600), Route is Internal
  Vector metric:
    Minimum bandwidth is 10000 Kbit
    Total delay is 2000 microseconds
    Reliability is 0/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 2
10.1.2.2 (Ethernet0), from 10.1.2.2, Send flag is 0x0
  Composite metric is (307200/281600), Route is Internal
  Vector metric:
    Minimum bandwidth is 10000 Kbit
    Total delay is 2000 microseconds
    Reliability is 0/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 2
```

## 設定に関する説明

**State is Passive** はネットワークがパッシブ状態にあることを意味します。つまり、このネットワークへのパスは検索の対象外です。安定したネットワークでは、ルートは必ずといってよいほどパッシブ状態にあります。

**Query origin flag is 1** : このフィールドは、このルートがアクティブの場合、このクエリーの発信者の情報を提供します。

0: このルートはアクティブですが、このルートに向けて発信されたクエリーはありません。(ローカルで到達可能後続を検索します。)

1: このルータが、このルートに対するクエリーを発信しました(またはルートがパッシブになっています)。

2: このクエリーに対する複数の拡散計算 このルータは複数の送信元から、このルートに対する複数のクエリーを受信しました。

3: このネットワークへのパスの学習元であるルータが、別のルートを問い合せていま

す。

4: このルートに対する複数のクエリー送信元があります ( このルートの学習時に経由したルータも含まれます )。2 と似ていますが、これはこのパスに対する未処理のクエリーについて記述した、クエリー送信元ストリングがあることも意味します。

**2 Successor(s)** は、このネットワークに到達可能なパスが 2 つあることを意味します。

**FD is 307200** は、このネットワークへの最適な現行メトリックを示します。ルートがアクティブの場合は、以前このネットワークにパケットをルーティングするために使用していたパスの、メトリックを示します。

**Routing Descriptor Blocks** 次のエントリはそれぞれ、ネットワークに通じる 1 つのパスについて説明したものです。

**10.1.1.2 (Ethernet1)** はネットワークへのネクストホップ、およびネクストホップに到着するために経由するインターフェイスです。

**from 10.1.2.2** はこのパス情報の送信元です。

**Send flag is:**

**0x0:** このエントリに関連して送信する必要があるパケットがあれば、そのパケットのタイプを示します。

**0x1:** このルータは、このネットワークへのクエリーを受信していて、ユニキャスト応答を送信する必要があります。

**0x2:** このルートはアクティブで、マルチキャストクエリーを送信する必要があります。

**0x3:** このルートは変更されており、マルチキャストアップデートを送信する必要があります。

**Composite metric is (307200/281600)** は、ネットワークへの合計計算コストを示します。括弧内の最初の数字は、このパスを経由した場合のネットワークへの合計コストで、ネクストホップへのコストも含まれます。括弧内の 2 番目の数字は、報告距離、つまり、ネクストホップが使用するコストです。

**Route is Internal** は、このルートがこの EIGRP AS ( 自律システム ) の中で配送元としていることを意味します。ルートがこの EIGRP AS に再配送された場合は、このフィールドはルートが External であることを示します。

**Vector metric** は、ネットワークへのコストを計算するために EIGRP が使用する個々のメト

リックを示します。EIGRP は、ネットワーク全体の総コスト情報を伝播しません。"the vector metrics are propagated, and each router computes the cost and reported distance individually."

**Minimum bandwidth is 10000 Kbit** は、このネットワークに向かうパスの最低帯域幅を示します。

**Total delay is 2000 microseconds** は、このネットワークに向かうパスの遅延の合計を示します。

**Reliability is 0/255** は信頼性係数を示します。この数字は動的に計算されますが、メトリック計算のデフォルトでは使用されません。

**Load is 1/255** は、リンクが伝送しているロードの量を示します。この数字は動的に計算され、EIGRP がこのパスの使用コストを計算する際には、デフォルトでは使用されません。

**Minimum MTU is 1500** このフィールドはメトリック計算では使用されません。

**Hop count is 2** これはメトリック計算では使用されませんが、EIGRP AS の最大サイズを制限します。EIGRP が許容する最大ホップ数はデフォルトで 100 ですが、最大数はメトリック最大ホップによって 220 に設定できます。

このルートが外部の場合には、次の情報が含まれます。次の表の下に、各出力フィールドの説明を示します。

外部ルート
External data: Originating router is 10.1.2.2 AS number of route is 0 External protocol is Static, external metric is 0 Administrator tag is 0 (0x00000000)

## 設定に関する説明

**Originating Router** は、これがこのルートを EIGRP AS に投入したルータであることを示しています。

**External AS** は、このルートの配布元である AS (ある場合) を示します。

**External Protocol** は、このルートの配布元であるプロトコル (ある場合) を示します。

**external metric** は、外部プロトコルの内部メトリックを示します。

**Administrator Tag** は、`set tag` および `match tag` コマンドでルートマップを使用して設定またはフィルタできます。



[show ip eigrp topology \[active | pending | zero-successors\]](#)

[show ip eigrp topology](#) と同じ出力形式ですが、トポロジ テーブルの一部も示します。

[show ip eigrp topology all-links](#)

[show ip eigrp topology](#) と同じ出力形式ですが、フィージブル サクセサだけではなく、トポロジ テーブルにあるすべてのリンクも表示されます。

## [関連情報](#)

- [EIGRP に関するサポートページ](#)
- [EIGRP コマンド リファレンス ガイド](#)
- [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とドキュメント - Cisco Systems](#)