IGRP の概要

内容

概要

ここでは Interior Gateway Routing Protocol (IGRP) の概要を示します。 このテクニカルノートには 2 つの目的があります。1 つは、IGRP のユーザや評価および実装を担当する方に IGRP テクノロジーを紹介することです。もう 1 つは、IGRP に組み込まれているいくつかの興味深い概念を広く公開することです。IGRP を設定する方法の詳細は、「Configuring IGRP」の「Cisco IGRP Implementation」、および「IGRP Commands」を参照してください。

IGRP の目的

IGRP は、多数のゲートウェイがそれぞれ自身のルーティングを調整できるようにするためのプロトコルです。IGRP の目標は次のとおりです。

- きわめて規模の大きいネットワークや複雑なネットワークでも安定したルーティングの実現。たとえ一時的であっても、ルーティング ループは発生しない。
- ネットワーク トポロジの変更に対する迅速な対応。
- 小さいオーバーヘッド。つまり、IGRP 自体はそのタスクにとって実際に必要とされる量よ

りも多くの帯域幅を使用しない。
- ほぼ同程度に望ましい複数のパラレル ルートがある場合に、それらのパラレル ルートの間でトラフィックを分割する。
- 異なるパスのエラー率とトラフィック レベルを考慮する。

IGRP の現在の実装は、TCP/IP のルーティングに対応しています。ただし、基本的にはさまざまなプロトコルを処理できるように設計されています。

それ 1 つですべてのルーティング問題を解決できるようなツールはありません。一般に、ルーティング問題は複数の部分に分かれます。IGRP などのプロトコルは「Internal Gateway Protocol」（IGP; 内部ゲートウェイ プロトコル）と呼ばれます。 IGP は、単一の管理体または緊密に連携した複数の管理体によって管理される 1 つのネットワーク セットの内部で使用することを目的としています。このようなネットワーク セットは「External Gateway Protocol」（EGP; 外部ゲートウェイ プロトコル）によって接続されます。 IGP は、ネットワーク トポロジに関する数多くの詳細情報を追跡管理するように設計されています。IGP の設計で優先されるのは、最適経路の生成と、変更への迅速な対応です。EGP の目的は、ネットワーク システムを、他のシステムによるエラーや意図的な情報の歪曲から保護することです。EGP の設計で優先されるのは、安定性と管理上の制御です。EGP では、ほとんどの場合、最適経路でなくても、妥当な経路が生成できれば十分用が足ります。

IGRP には、Xerox の Routing Information Protocol、Berkeley の RIP、Dave Mills の Hello といった旧プロトコルとの間にいくつかの類似点があります。これらのプロトコルとの主な相違点は、IGRP が大規模で複雑なネットワーク向けに設計されている点です。セクション 4 に RIP との詳細な比較があります。

これらの旧プロトコルと同様に、IGRP もディスタンス ベクター プロトコルです。ディスタンス ベクター プロトコルでは、ゲートウェイは隣接ゲートウェイとのみルーティング情報を交換します。このルーティング情報には、残りのネットワークに関する情報の要約が含まれています。すべてのゲートウェイがこれらの情報を総合すれば、結果的に分散アルゴリズムが実行されることになり、これによって最適化問題を解決できることが数学的に証明できます。各ゲートウェイは問題の一部分のみを解決し、さらに全データの一部分のみを受信するだけで済みます。

もう 1 つの大きな選択肢として、Shortest-Path First（SPF）と呼ばれるアルゴリズムの種類があります。OSPF はこの概念を使用します。OSPF に関する詳細は、「OSPF 設計ガイド」を参照してください。これらのアルゴリズムはフラッディング手法に基づいており、すべてのゲートウェイが他のすべてのゲートウェイ上にあるすべてのインターフェイスの最新ステータスを保持します。各ゲートウェイは、ネットワーク全体のデータを使用して、自身の視点から最適化問題を単独で解決します。どちらのアプローチにもそれぞれ長所があります。ある状況では、SPF の方が変更に迅速に対応できます。IGRP でルーティング ループを避けるためには、ある種の変更が起こった後、数分間新しいデータを無視する必要があります。SPF は各ゲートウェイから直接情報を受け取るため、このようなルーティング ループを回避できます。そのため、新しい情報に基づいて即時に動作を開始できます。ただし、SPF では内部データ構造とゲートウェイ間のメッセージに含まれるデータ量が IGRP よりもかなり多く、それらの大量のデータを処理する必要があります。

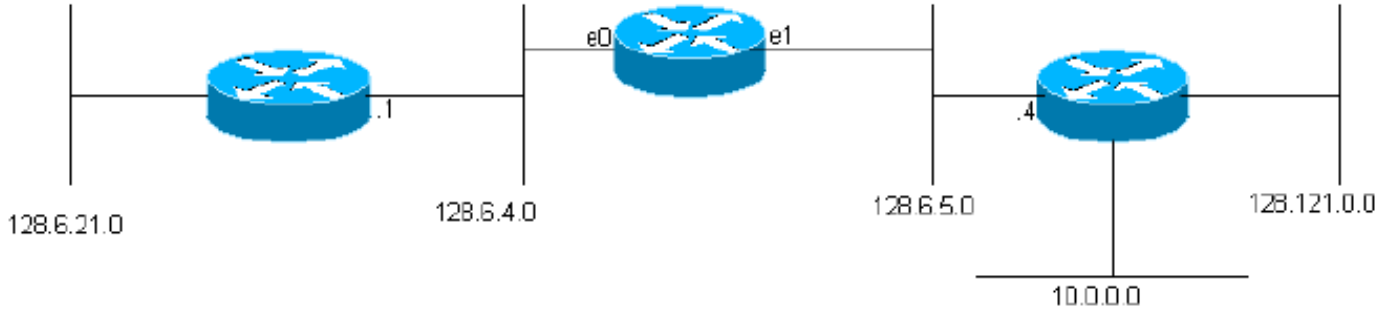
ルーティングに関する問題

IGRP は、複数のネットワークに接続したゲートウェイで使用することを目的としています。それらのネットワークではパケットベースのテクノロジーが使用されていることを想定しています。実際にゲートウェイはパケット スイッチとして機能します。あるネットワークに接続されているシステムが別のネットワーク上にあるシステムにパケットを送信しようとするとき、このシステムはパケットをゲートウェイ宛てに送信します。パケットの宛先がゲートウェイに接続されて

いるネットワークの 1 つであった場合、ゲートウェイはパケットを宛先に転送します。パケットの宛先がそれよりも遠い場合、ゲートウェイは宛先に近い位置にある別のゲートウェイにパケットを転送します。ゲートウェイは、パケットの処理方法を決定する際にルーティング テーブルを利用します。次に、簡単なルーティング テーブルの例を示します。(例で使用されているアドレスはラトガース大学から取得した IP アドレスです。基本的なルーティング問題は他のプロトコルでも同様ですが、ここでは IP ルーティングのために IGRP を使用していることを前提とします)。

図 1：

```
network       gateway      interface
  -------       -------      ---------
  128.6.4       none         ethernet 0
  128.6.5       none         ethernet 1
  128.6.21      128.6.4.1    ethernet 0
  128.121       128.6.5.4    ethernet 1
  10            128.6.5.4    ethernet 1
```



(実際の IGRP ルーティング テーブルには、各ゲートウェイに関するその他の情報も含まれています。これについては後述します。) このゲートウェイは、2 つのイーサネット、0 と 1 に接続されています。これらのイーサネットにはそれぞれ、128.6.4 および 128.6.5 という IP ネットワーク番号 (実際はサブネット番号) が割り当てられています。したがって、これら特定のネットワーク宛てのパケットは、単純に対応するイーサネット インターフェイスを使用して、宛先に直接送信できます。128.6.4.1と128.6.5.4の2つの近接ゲートウェイがあります。128.6.4と128.6.5以外のネットワークのパケットは、これらのゲートウェイの1つまたはもう1つに転送されます。ルーティング テーブルは、どのネットワークに対してどちらのゲートウェイを使用すればよいかを示しています。たとえば、ネットワーク10上のホスト宛てのパケットは、ゲートウェイ128.6.5.4に転送する必要があります。このゲートウェイはネットワーク10に近い場所にあり、ネットワーク10へのベストパスがこのゲートウェイを通過することを望んでいます。IGRP の第 1 の目的は、このようにゲートウェイがルーティング テーブルを作成し、メンテナンスできるようにすることです。

IGRP の要約

前述したように、IGRP は、各ゲートウェイが他のゲートウェイと情報を交換することで、固有のルーティング テーブルを作成できるようにするためのプロトコルです。ゲートウェイは最初に、自身に直接接続されているすべてのネットワークのエントリを作成します。次に、隣接ゲートウェイとルーティング更新を交換して、他のネットワークに関する情報を取得します。最も単純なケースでは、ゲートウェイは各ネットワークに到達するための最適な経路を表すパスを 1 つ見つけ出します。パスにはそれぞれ、パケットを次に送信するゲートウェイ、使用するネットワーク インターフェイス、メトリック情報などの特性が記述されています。メトリック情報とは、そのパスがどれだけ適しているかを示す数値のセットです。ゲートウェイは、このメトリック情報を使用して、数多くのゲートウェイから受信したパスを比較し、使用するパスを決定します。メ

トリック情報は、2 つ以上のパスの間でトラフィックを分割する際にも利用されます。IGRP では、2 つ以上のパスの適切度が等しい場合は必ずトラフィックが分割されます。複数のパスの適切度がほぼ等しい場合のトラフィックの分割方法をユーザが設定することもできます。この場合は、パスのメトリックが適切であるほど、そのパスに従って送信されるトラフィック量が多くなります。これは 9600 bps 回線と 19200 bps 回線の間でトラフィックを分割すると、19200 bps 回線のトラフィック量が 9600 bps 回線のトラフィック量のほぼ 2 倍になることを意図しています。

IGRP では次のようなメトリックが使用されます。

- トポロジ上の遅延時間
- パスの中で最も帯域幅の狭いセグメントの帯域幅
- パスのチャネル占有率
- パスの信頼性

トポロジ上の遅延時間とは、ネットワークに負荷のない状態で、そのパスに従って宛先に到達するまでにかかる時間の長さです。ネットワークに負荷がかかっている場合は、当然その他の遅延が存在します。ただし負荷は、実際の遅延を測定するのではなく、チャネル占有率を使用して算出されます。パス帯域幅とは単純に、そのパスの中で最も低速なリンクの帯域幅（bps）のことです。チャネル占有率は、その帯域幅が現在どの程度使用されているかを示します。チャネル占有率は測定され、負荷に応じて変わります。信頼性は現在のエラー率を示します。これは宛先に到達したパケットの割合です。この値は測定されます。

メトリックの一部としては使用されないものの、メトリックとともに渡される情報があと 2 つあります。ホップ カウントと MTU です。ホップ カウントとは単純に、パケットが宛先に到達するために通過しなければならないゲートウェイの数です。MTU は、パス全体を通じて分割せずに送信できる最大のパケットサイズです（つまり、そのパスに関係するすべてのネットワークの中で最小の MTU を表します）

メトリック情報に基づいて、そのパスに対する「複合メトリック」が 1 つだけ計算されます。複合メトリックとは、各種メトリック コンポーネントの影響を合計して、そのパスの「適切度」を表す 1 つの数値にしたものです。ベスト パスを決定する際に実際に使用されるのは、この複合メトリックです。

各ゲートウェイは、すべての隣接ゲートウェイに対して自身のルーティング テーブル全体（スプリット ホライズン規則で使用する検閲情報を含む）を定期的にブロードキャストします。他のゲートウェイからこのブロードキャストを受信したゲートウェイは、受け取ったテーブルと既存のテーブルを比較します。新しい宛先とパスがあればゲートウェイのルーティング テーブルに追加します。ブロードキャストに含まれるパスは既存のパスと比較されます。新しいパスの方が適している場合は、そのパスに既存のパスが置き換えられます。さらに、ブロードキャスト中の情報を使用して、既存のパスに関するチャネル占有率などの情報を更新します。この基本的な手順は、すべてのディスタンス ベクター プロトコルで使用される手順とほぼ同じです。これは、数学関係の文献ではベルマンフォード アルゴリズムと呼ばれています。従来のディスタンスベクタープロトコルであるRIPを説明する基本手順の詳細な開発については 、RFC 1058を参照してください。

IGRP では、一般的なベルマンフォード アルゴリズムに対して、3 つの重要な側面で変更を加えています。第 1 に、単純なメトリックの代わりに、メトリックのベクターを使用してパスの特性を記述しています。第 2 に、最小のメトリックを持つ 1 つのパスを選択する代わりに、メトリックが指定された範囲内にある複数のパスの間でトラフィックを分割します。第 3 に、トポロジの変更が伝搬されている状況でネットワークを安定させるための機能がいくつか追加されています。

最適パスは、次の複合メトリックに基づいて選択されます。

```
[(K1 / Be) + (K2 * Dc)] r
```

K1、K2 = 定数 Be = 無負荷状態でのパスの帯域幅 x (1 - チャネル占有率) Dc = トポロジ上の遅延 r = 信頼性

複合メトリックが最小のパスが最適パスになります。1 つの宛先へのパスが複数ある場合、ゲートウェイは複数のパスを通じてパケットをルーティングできます。これは、各データ パスの複合メトリックに従って行われます。たとえば、複合メトリックが 1 のパスと複合メトリックが 3 のパスがある場合、複合メトリックが 1 のデータ パスを通じて 3 倍の量のパケットが送信されます。

メトリック情報のベクターを使用する方法には、2 つの長所があります。第 1 に、1 つのデータ セットを使用して複数のタイプ オブ サービスをサポートできます。第 2 に、精度が向上します。単一のメトリックを使用する場合、一般にそれは遅延として扱われます。パス内の各リンクがメトリックの合計に加算されます。低帯域幅のリンクがある場合、通常は大きな遅延となって表れます。ただし、帯域幅の制限は、実際に遅延のように累積はされません。帯域幅を別個のコンポーネントとして扱えば、正しく処理できます。同様に、負荷は異なるチャネル占有率によって処理できます。

IGRP は、ループを含む一般的なグラフ トポロジを安定して処理できる、コンピュータ ネットワークを相互接続するための仕組みを提供します。このシステムでは、完全なパス メトリック情報が管理されます。つまり、ゲートウェイが接続されている他のすべてのネットワークへのパス パラメータがわかっています。そのため、トラフィックを複数のパラレル パスにわたって分散することが可能となり、さらに複数のパス パラメータをネットワーク全体にわたって同時に計算できます。

RIP との比較

この項では、IGRP と RIP を比較します。RIP は IGRP と同様の目的で幅広く使用されているため、この比較は有益です。ただし、この比較は完全に公平なものではありません。RIP は、IGRP と同じ目標をすべて満たすことを目的としたものではありませんでした。RIP は、テクノロジーにある程度一貫性のある小規模ネットワークでの使用を目的としたものです。このような用途では、一般に RIP が適しています。

IGRP と RIP との間の最も基本的な相違はメトリックの構造です。残念ながら、この変更は RIP に後から簡単に追加できません。IGRP で採用されている新しいアルゴリズムとデータ構造が必要です。

RIP は、ネットワークを記述するために単純な「ホップ カウント」メトリックを使用します。すべてのパスが遅延や帯域幅などによって記述される IGRP とは異なり、RIP ではパスが 1～15 の数値で記述されます。通常、この数値は、パスが宛先に到達するまでに通過するゲートウェイの数を表します。これは、低速のシリアル回線とイーサネットの間にまったく区別がないことを意味します。RIP の実装の中には、システム管理者が、特定のホップを複数回カウントするように指定できるものもあります。低速のネットワークは大きなホップ カウントで表現されます。ただし、最大値が 15 であるため、この方法でもそれほど多くの状況に対応できません。たとえば、イーサネットを 1、56 Kb 回線を 3 で表す場合、1 つのパスには多くても 5 つの 56 Kb 回線しか含めることができません。そうしないと、最大値の 15 を超えてしまいます。シスコの研究によれば、大規模なネットワークを考慮しながら、使用可能なネットワーク速度の範囲全体を表現するには、24 ビットのメトリックが必要とされます。最大メトリックがあまりにも小さすぎる場合、システム管理者には不本意な選択肢しかありません。つまり、高速な経路と低速な経路を区別

できないか、またはネットワーク全体を制限内に収めることができません。実際に、全国的なネットワークの多くはすでに、すべてのホップを 1 回だけカウントしたとしても RIP では処理できないほど規模が大きくなっています。RIP は、これらのネットワークではまったく使用できません。

これに対してすぐに思いつく対応策は、RIP を修正してより大きいメトリックを使用できるようにすることです。残念ながら、これはうまくいきません。すべてのディスタンス ベクター プロトコルと同様に、RIP も「無限カウント」の問題を抱えています。 詳細はRFC 1058を参照してください 。トポロジが変化すると、誤った経路がアドバタイズされます。これらの誤った経路に対応するメトリックは 15 に達するまでゆっくりと増加し、それを超えた時点で経路は削除されます。トリガ更新が使用されるとすれば、15 という最大値は十分に小さいので、このプロセスはかなり迅速に収束します。RIP を修正して 24 ビットのメトリックを使用できるようにすると、メトリックが最大で 2**24 回カウントされるまでループが存続することになります。これは許容できません。IGRP は、誤った経路がアドバタイズされることを防ぐために設計された機能を備えています。これらはセクション5.2で説明します。これらの機能を導入したり、SPFなどのプロトコルに変更したりせずに、複雑なネットワークを処理することは現実的ではありません。

IGRP では、単に許容できるメトリックの範囲が拡大しただけではありません。メトリックの構造が見直され、遅延、帯域幅、信頼性、および負荷を記述できるようになっています。RIP のような単一のメトリックでもこれらの特性を表現することは可能ですが、IGRP で採用されたアプローチの方が、精度が高いといえます。たとえば、単一メトリックでは、複数の連続した高速リンクが 1 つの低速リンクと等価に見えます。これは、遅延が最も重要である対話型トラフィックにとって問題となる可能性があります。ただし、バルク データ転送では、最も重要となるのは帯域幅であり、メトリックを合計することは適切なアプローチではありません。IGRP は遅延と帯域幅を別々に処理し、遅延は累積しますが、帯域幅は最小値をとります。信頼性と負荷の影響を単一コンポーネント メトリックにどのように組み込めばよいか理解することは容易ではありません。

筆者の意見では、IGRP の大きな長所の 1 つは構成の容易さです。IGRP では、物理的な意味を持つ量を直接表現できます。つまり、IGRP はインターフェイス タイプや回線速度に基づいて自動的に設定できます。単一コンポーネント メトリックでは、複数の異なるものの影響を組み込むには、多くの場合、メトリックを「加工」する必要があります。

その他の新機軸は、ルーティング プロトコルというよりもアルゴリズムとデータ構造に関連しています。たとえば、IGRP の仕様には、複数経路間でのトラフィックの分割をサポートするアルゴリズムとデータ構造が含まれています。確かに、それと同じことを行う RIP の実装を設計することは可能です。しかし、ルーティングが再実装された後で、RIP を使用し続ける理由はありません。

ここまでは、「汎用的な IGRP」、つまり任意のネットワーク プロトコルのルーティングをサポートできるテクノロジーについて記述してきました。しかし、このセクションでは、TCP/IP 固有の実装について少し触れることにも意味があります。以降では、TCP/IP 実装と RIP を比較します。

RIP 更新メッセージには、単純にルーティング テーブルのスナップショットが含まれます。つまり、このメッセージには数多くの宛先とメトリック値が含まれており、それ以外の情報はほとんどありません。IGRP の IP 実装では、それらとは異なる構造が追加されています。第 1 に、更新メッセージが「自律システム番号」で識別されます。 この用語は Arpanet の伝統から生まれたもので、そこでは固有の意味を持っています。しかし、ほとんどのネットワークにとってその意味するところは、「1 つのネットワーク上で複数の異なるルーティング システムを実行できる」ということです。これは、複数の組織からのネットワークが集中している場所で役立ちます。各組織は、それぞれ固有のルーティングを維持できます。それぞれの更新にはラベルが付けられるため、ゲートウェイが正しい更新にのみ注意を払うように設定できます。一部のゲートウェイは、

複数の自律システムから更新を受信するように設定されます。これらのゲートウェイは、制御された方法に従ってシステム間で情報を受け渡します。これはルーティング セキュリティの問題に対する完全なソリューションではありません。どのゲートウェイもすべての自律システムからの更新を受信するように設定できます。しかしそれでも、ネットワーク管理者の間に相応の信頼性があるところでルーティング ポリシーを実装する上では、これは非常に有効なツールです。

IGRP 更新メッセージに関する第 2 の構造上の特徴は、IGRP によるデフォルト ルートの処理方法に関係します。ほとんどのルーティング プロトコルはデフォルト ルートの概念を持っています。ルーティング更新に世界中の全ネットワークのリストを含めるのは、通常は現実的ではありません。一般にゲートウェイのセットは、組織の内部にあるネットワークの詳細なルーティング情報を必要とします。組織の外部にある宛先へのトラフィックはすべて、数台ある境界ゲートウェイの中の 1 台に送信できます。これらの境界ゲートウェイには、他のゲートウェイよりも詳細な情報が保持されます。最適な境界ゲートウェイへの経路が「デフォルト ルート」です。 ここでの「デフォルト」は、「内部ルーティング更新に明示的に含まれていないすべての宛先に到達するために使用される」ことを意味します。RIP と他のルーティング プロトコルの一部は、デフォルト ルートに関する情報を実際のネットワークと同様に累積します。IGRP のアプローチはそれとは異なります。IGRP では、デフォルト ルート用に単一の擬似エントリを作成するのではなく、実際のネットワークに対して、デフォルトとなりうる候補としてのフラグを設定できます。これを実装するには、更新メッセージの特別な外部セクションに、候補となるネットワークに関する情報を設定します。ただしこれは、これらのネットワークに対応するビットをオンにすると考えた方がいいでしょう。IGRP は、デフォルト ルートの候補すべてを定期的にスキャンし、実際のデフォルト ルートとして、メトリックが最も小さいものを選択します。

デフォルトに対するこのアプローチは、ほとんどの RIP 実装で採用されているアプローチよりもいくぶん柔軟性が高いといえます。一般に RIP ゲートウェイは、特定のメトリックを持つデフォルト ルートを生成するように設定できます。このデフォルト ルートの生成は、境界ゲートウェイで行われることを意図しています。

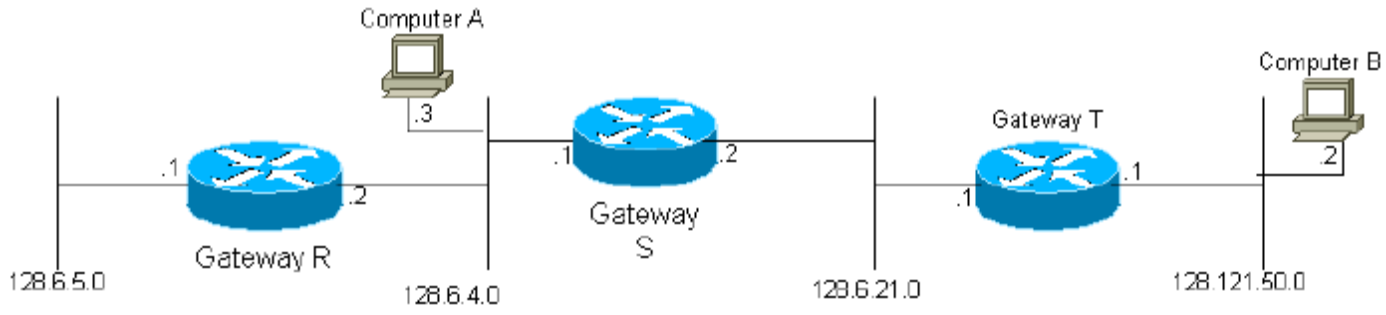
詳細な説明

この項では、IGRP について詳しく説明します。

全体説明

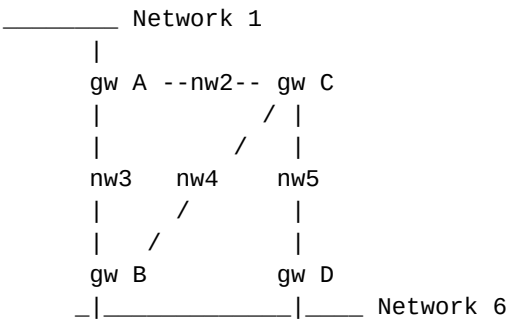
ゲートウェイに最初に電源を投入すると、ルーティング テーブルが初期化されます。この初期化は、コンソール端末からオペレータが実行することも、コンフィギュレーション ファイルから情報を読み込んで実行することもできます。初期化によって、ゲートウェイに接続された各ネットワークの記述が与えられます。これには、リンクに従ったトポロジ上の遅延（1 つのビットがリンクを通過するのにかかる時間）やリンクの帯域幅などが含まれます。

図 2



たとえば、図 1 のゲートウェイ S には、対応するインターフェイスを通じてネットワーク 2 と 3 に接続されていることが通知されます。したがって、ゲートウェイ2はネットワーク2と3の任意の宛先コンピュータに到達できることを初めて認識します。すべてのゲートウェイは、初期化された情報と他のゲートウェイから収集された情報を定期的に隣接ゲートウェイに送信するようにプログラムされます。ゲートウェイSはゲートウェイRとTから更新を受信し、ネットワーク1のコンピュータにゲートウェイRを介して、ネットワーク4のコンピュータにゲートウェイTを介して到達できることを学習します完全に接続されています

図 3



各ゲートウェイは、複合メトリックを計算して、宛先コンピュータへのデータ パスがどれだけ望ましいかを判断します。たとえば、図 2 で宛先がネットワーク 6 上にある場合、ゲートウェイ A はゲートウェイ B と C を経由する 2 つのパスのメトリック関数を計算します。パスは単純にネクストホップによって定義されている点に注意してください。A からネットワーク 6 への実行可能な経路は、実際には 3 つあります。

- 直接 B へ
- C、B の順に経由
- C、D の順に経由

ただし、ゲートウェイ A は C に関係する 2 つの経路のどちらかを選択する必要はありません。A のルーティング テーブルには、C へのパスを表すエントリは 1 つしかありません。そのメトリックは、C から最終的な宛先へ到達する最適な経路を表しています。A が C にパケットを送信した場合、B と D のどちらを使用するかは C が判断します。

式 1 :

各データ パスについて計算される複合メトリック関数は次のとおりです。

```
[(K1 / Be) + (K2 * Dc)] r
```

r = 信頼性（割合）（ネクストホップで正常に受信される伝送の %）、Dc = 複合遅延、Be = 実効帯域幅：無負荷状態での帯域幅 X（1 - チャネル占有率）、K1 および K2 = 定数

式 2 :

原則的には、複合遅延（Dc）は次のように算出できます。

```
Dc = Ds + Dcir + Dt
```

Ds = スイッチング遅延 Dcir = 回線遅延（1 ビットの伝搬の遅延）DT = 伝送遅延（1500 ビット メッセージの無負荷状態での遅延）

ただし、実際には、ネットワーク テクノロジーのタイプ別に標準遅延値が使用されます。たとえば、イーサネットやシリアル回線に対して特定のビット レートの標準遅延値があります。

図 2 のケースにおけるゲートウェイ A のルーティング テーブルの例を次に示します。(簡単にするために、メトリック ベクターの個々のコンポーネントは示されていません)

Routing Table Example:

Network	インターフェイス	次のゲートウェイ	測定項目
1	NW 1	なし	直接接続
0	NW 2	なし	直接接続
3	NW 3	なし	直接接続
4	NW 2	C	1270
	NW 3	B	1180
5	NW 2	C	1270
	NW 3	B	2130
6	NW 2	C	2040
	NW 3	B	1180

近接ゲートウェイと情報を交換することでルーティング テーブルを作成する基本的なプロセスは、ベルマンフォード アルゴリズムで説明されます。このアルゴリズムは、RIP (RFC 1058) などの旧プロトコルでも使用されています。 さらに複雑なネットワークを取り扱うために、IGRP では基本的なベルマンフォード アルゴリズムに 3 つの機能が追加されています。

1. 単純なメトリックの代わりに、メトリックのベクターを使用してパスの特性を記述します。このベクターから、式 1 に従って 1 つの複合メトリックを計算できます。ベクターを使用すると、ゲートウェイは、式1の複数の異なる係数を使用して、異なるタイプのサービスに対応できます。また、ネットワークの特性を単一のメトリックよりも正確に表現することもできます。
2. 最小のメトリックを持つ 1 つのパスを選択する代わりに、メトリックが指定された範囲内にある複数のパスの間でトラフィックを分割します。これにより、複数の経路を同時に使用できるので、単一の経路を使用する場合よりも帯域幅を有効に利用できます。バリアンス値 (V) はネットワーク管理者が指定します。最小の複合メトリック (M) を持つパスすべてが確保されます。さらに、メトリックが V x M より小さいパスすべてが確保されます。トラフィックは、複数のパスの間で複合メトリックに反比例するように分散されます。
3. このバリアンスの概念にはいくつか問題があります。1 を超えるバリアンス値を使用しながら、なおかつパケットがループしないようにする方法を見つけ出すことが困難です。シスコ リリース 8.2 では、バリアンス機能は実装されていません。(この機能が削除されたリリース不明) これにより、バリアンスは常に 1 に設定されるようになりました。
4. トポロジが変化する状況でネットワークを安定させるための機能がいくつか追加されています。これらの機能の目的は、ルーティング ループと「無限カウント」を防ぐことです。この問題は、この種の用途に対してフォードタイプのアルゴリズムを使用する、従来の試みの特性でした。主な安定性機能は、「ホールドダウン」、「トリガ更新」、「スプリット ホライズン」、および「ポイズニング」です。 これらについては、後で詳しく説明します。

トラフィック分割 (第 2 項) を使用すると、潜在的な危険性がかなり向上します。バリアンス V は、ゲートウェイが速度の異なるパラレル パスを使用できるようにする目的で開発されました。たとえば、冗長化のために 9600 bps 回線と 19200 bps 回線を並行して運用しているとします。バリアンスが 1 の場合は、最適パスのみが使用されます。そのため、19200 bps 回線に相応の信

頼性があれば、9600 bps 回線は使用されません (ただし、同一のパスが存在する場合、その口ードはそれらのパスの間で共有されます)。 バリアンスを大きくすると、最適経路と、ほぼ同程度に適切なその他の経路の間でトラフィックを分割できます。バリアンスを十分大きい値にすると、トラフィックはこの 2 つの回線の間で分割されます。危険性とは、バリアンスを十分大きい値にすると、パスが低速になるだけでなく、実際に「間違った方向」になることです。 そのため、トラフィックが「上流」に送信されることを防ぐために規則を追加する必要があります。リモートの複合メトリック (ネクストホップで計算された複合メトリック) がゲートウェイで計算された複合メトリックよりも大きくなるようなパスに従ってトラフィックが送信されることはありません。一般に、パラレル パスを使用しなければならない特定の状況を除き、1 より大きいバリアンスを設定しないことをお勧めします。パラレル パスを使用する場合は、「正しい」結果が得られるようにバリアンスを注意深く設定するようにします。

IGRP は、複数の「タイプ オブ サービス」と複数のプロトコルを取り扱うことを目的としています。タイプ オブ サービスはデータ パケット内の仕様の 1 つで、これによってパスの評価方法が変わります。たとえば TCP/IP プロトコルでは、パケットによって高帯域幅、低遅延、高信頼性といった相対的な重要度を指定できます。一般に、対話型アプリケーションでは低遅延を指定し、バルク転送アプリケーションでは高帯域幅を指定します。これらの要件によって、式 1 で使用される適切な K1 と K2 の相対値が決まります。1.サポートされるパケット内の仕様の各組み合わせは、「タイプオブサービス」と呼ばれます。 タイプ オブ サービスごとに、パラメータ K1 と K2 のセットを選択する必要があります。ルーティング テーブルはタイプ オブ サービスごとに保持されます。これは、式 1 で定義された複合メトリックに従ってパスが選択され、パスの優先順位が決まるためです。1.これは、サービスのタイプごとに異なります。これらすべてのルーティング テーブルからの情報を集約して、ゲートウェイで交換されるルーティング更新メッセージが生成されます (図 7 を参照)。

安定性機能

この項では、ホールドダウン、トリガー更新、スプリット ホライズン、およびポイズニングについて説明します。これらの機能が設計された目的は、ゲートウェイが誤った経路を選択しないようにすることです。RFC 1058で説明されているように 、これはゲートウェイまたはネットワークの障害が原因でルートが使用不能になったときに発生する可能性があります。原則的には、隣接ゲートウェイが障害を検出します。障害を検出したゲートウェイは、その使用できなくなった経路を使用不能として通知するルーティング更新を送信します。しかし、その更新がネットワークの一部にまったく到達しなかったり、あるいは特定のゲートウェイへの到達が遅れたりする場合があります。その間に、使用できなくなった経路がまだ良好であると信じているゲートウェイがその情報を引き続き伝搬することで、障害経路が再びシステムに注入される可能性があります。やがてこの情報はネットワークを通じて伝搬され、それを再注入したゲートウェイに戻ります。その結果、循環経路が発生します。

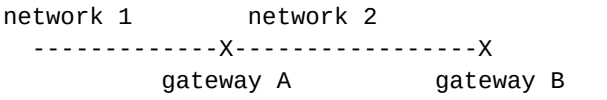
実際のところ、この対応策には冗長性を持たせてあります。原則的に、ホールドダウンとトリガー更新は、最初の段階で誤った経路が伝搬しないようにするためには十分役立ちます。しかし、実際には、さまざまな種類の通信障害が発生する可能性があるため、これらの機能だけでは十分対応できません。スプリット ホライズンとルート ポイズニングは、どのような場合でもルーティング ループが発生しないようにすることを目的としています。

通常は、新しいルーティング テーブルが近接ゲートウェイに定期的に送信されます (デフォルトでは 90 秒間隔ですが、この値はシステム管理者が調整できます)。トリガー更新は、なんらかの変更に応じて即時に送信される新しいルーティング テーブルです。最も重要な変更は経路の削除です。これは、タイムアウトが発生した場合 (おそらく、近接ゲートウェイまたは回線がダウンしている)、またはパス上の次のゲートウェイから受信した更新メッセージによってパスの使用不能が通知された場合に起こります。ゲートウェイ G は、ある経路が使用できないことを検出すると、即時に更新をトリガーします。この更新は、その経路を使用不能として通知します。こ

の更新が近接ゲートウェイに到達したときにどうなるかを考えてみましょう。近接ゲートウェイの経路が G の方向を指している場合、近接ゲートウェイはその経路を削除する必要があります。そのため、近接ゲートウェイは更新などをトリガーします。このように、ひとたび障害が発生すると更新メッセージの波が起こります。この波は、障害が発生したゲートウェイまたはネットワークを経由した経路がそれまでに到達していたネットワーク部分の全体にわたって伝搬します。

更新の波が該当するすべてのゲートウェイに即時に到達することが保証できれば、トリガー更新で十分です。しかし、問題が 2 つあります。第 1 に、更新メッセージを含むパケットが廃棄されたり、ネットワーク内のリンクによって破壊されたりする可能性があります。第 2 に、トリガー更新は瞬時には起こりません。トリガー更新をまだ受信していないゲートウェイがちょうど悪いタイミングで定期更新を発行し、すでにトリガー更新を受信した近接ゲートウェイに障害経路が再び挿入される可能性があります。ホールドダウンは、これらの問題を回避するために設計されています。ホールドダウン規則では、経路が削除されると、同じ宛先への新しい経路が一定時間受け入れられません。これによってトリガー更新が他のすべてのゲートウェイに到達するための時間的余裕が生まれ、取得した経路が、ゲートウェイによって再び挿入された障害経路でないことを保証できます。ホールドダウン時間は、トリガー更新の波がネットワーク全体に行きわたるまでの時間を考慮して十分長くする必要があります。また、廃棄されたパケットに対処するため、数回の定期的なブロードキャスト サイクルを含めることも必要です。トリガー更新の 1 つが廃棄されるか、破損した場合にどうなるかを考えてみましょう。その更新を発行したゲートウェイは、次の定期更新時に別の更新を発行します。これによって、最初の波を逸した近接ゲートウェイからトリガー更新の波が再び始まります。

タイムアウトした経路を取り除き、その経路が再び挿入されないようにするには、トリガー更新とホールドダウンの組み合せで十分です。しかし、いずれにしても追加の予防策は実施する意味があります。これらの対策は、非常に損失の大きいネットワークや分割されたネットワークも考慮しています。IGRP で必要とされる追加の予防策は、スプリット ホライズンとルート ポイズニングです。スプリット ホライズンは、経路をその送信元方向に返送することには意味がないという考えから生まれました。次の状況について考えてみます。



ゲートウェイ A はゲートウェイ B に対して、ネットワーク 1 への経路を通知します。B が A に更新を送信するときに、ネットワーク 1 の情報を含める理由はありません。また、A は B よりも 1 に近いため、A が B を経由することを考慮する理由もありません。スプリット ホライズン規則では、近接ゲートウェイごと (実際は近接ネットワークごと) に異なる更新メッセージが生成されます。 特定の近接ゲートウェイ用の更新では、その近接ゲートウェイを指す経路が省略されます。このルールは隣接ゲートウェイ間のループを防止します。たとえば、ネットワーク 1 への A のインターフェイスが故障したと仮定します。スプリットホライズンルールがないと、BはAに1に到達できることを通知します。Aは実際のルートを持っていないため、Aがそのルートをピックアップする可能性があります。この場合、AとBの両方に1へのルートがありますが、AはBをポイントし、BはAをポイントします。もちろん、トリガーされたアップデートとホールドダウンは、この問題の発生を防止する必要があります。しかし、送信元に情報を返送する理由はないため、いずれにしてもスプリット ホライズンは実行する意味があります。スプリット ホライズンには、ループを防止するという役割のほかに、更新メッセージのサイズを抑える効果もあります。

スプリット ホライズンは隣接ゲートウェイ間のループを防止します。ルート ポイズニングの目的は、それよりも大きなループを除去することです。ルート ポイズニング規則では、更新に含まれている既存ルートのメトリックがある条件を満たすほど増加した場合はループが存在するとされます。その経路は削除され、ホールドダウン状態になります。現在、複合メトリックの増加が1.1倍を超えるとルートが削除されます。チャネル占有率や信頼性の変化によりメトリックの小さな変更が発生する可能性があるため、複合メトリックの増加だけでは安全ではありません。その

ため、1.1 という増加率は発見を助けるだけです。正確な値は重要ではありません。小さいループはトリガー更新とホールドダウンで防ぐことができるため、ルート ポイズニング規則が必要となるのはそれよりも大きいループを除去する場合のみであると考えられます。

ホールドダウンの無効化

リリース 8.2 の時点で、シスコのコードにはホールドダウンを無効にするオプションが用意されています。ホールドダウンの短所は、古い経路で障害が発生したときに新しい経路の受け入れが遅れる点です。デフォルト パラメータでは、経路の変更後、ルータが新しい経路を受け入れるまでに数分間かかる場合があります。しかし、上記のような理由から、単純にホールドダウンを無効にするのは安全ではありません。結果として、RFC 1058 に説明されているように、無限カウントが発生する恐れがあります。ルート ポイズニングのより強力なバージョンを使用すれば、無限カウントを防ぐ上でホールドダウンは不要になると推測されます (ただし、実証はできません) 。そのため、ホールドダウンを無効にすると、ルート ポイズニングの強力なバージョンが有効になります。ただし、スプリット ホライズンとトリガー更新は引き続き有効です。

ルート ポイズニングの強力なバージョンはホップ カウントに基づきます。あるパスのホップ カウントが増えると、経路が削除されます。この方法では明らかに、有効な経路も削除されます。ネットワーク内のどこかでパスの通過するゲートウェイの数が 1 つ増えるような変更が起こると、ホップ カウントは増えます。このケースでは、経路はまだ有効です。しかし、このケースとルーティング ループ (無限カウント) を識別するための、まったく安全な方法はありません。 したがって、最も確実なアプローチは、「ホップ カウントが増えた場合は必ず経路を削除する」ということになります。削除された後も正当であれば、その経路は次の更新によって再び挿入され、続いてトリガー更新が引き起こされてシステム全体に再び挿入されます。

一般に、ディスタンス ベクター アルゴリズムは新しい経路を容易に受け入れます。問題はシステムから古い経路を一掃することです。したがって、疑わしい経路を大胆に削除する規則は安全であるといえます。

アップデートプロセスの詳細

図 4 ~ 8 に示す一連のプロセスは、1 つのネットワーク プロトコル、たとえば TCP/IP、DECnet、ISO/OSI プロトコルなどを処理することを目的としています。ただし、ここでは TCP/IP についてのみ、プロトコルの更新プロセスの詳細を説明します。1 台のゲートウェイで、複数のプロトコルに準拠したデータを処理する場合があります。プロトコルのアドレッシング構造とパケット フォーマットはそれぞれ異なるため、図 4 ~ 8 の実装に使用されるコンピュータ コードは、通常、プロトコルごとに異なります。図 4 に関する説明で詳しく述べるとおり、図 4 のプロセスが最も異なります。図 5 ~ 8 のプロセスでは、大まかな構造はほぼ同じです。プロトコル間での主な相違点はルーティング更新パケットのフォーマットで、これは特定のプロトコルにあわせて設計する必要があります。

宛先の定義がプロトコル間で異なる場合があります。ここで示す方法は、個々のホストやネットワークへのルーティング、またはより複雑な階層アドレス方式に使用できます。どのタイプのルーティングを使用するかは、プロトコルのアドレッシング構造によって決まります。現在の TCP/IP 実装では、IP ネットワークへのルーティングのみがサポートされています。したがって、「宛先」は、実際には IP ネットワークまたはサブネット番号を意味します。サブネット情報は、接続されたネットワークに関するもののみが保持されます。

図 4 ~ 7 は、ゲートウェイで使用されるルーティング プロセスのさまざまな断片を示す疑似コードです。プログラムの最初に、受け入れ可能なプロトコルと、各インターフェイスを記述するパラメータが入力されます。

ゲートウェイは、リストされている特定のプロトコルのみを処理します。リスト上にないプロトコルを使用したシステムからの通信は無視されます。入力データは次のとおりです。

- ゲートウェイが接続されているネットワーク
- 各ネットワークの無負荷状態での帯域幅
- 各ネットワークのトポロジ上の遅延
- 各ネットワークの信頼性
- 各ネットワークのチャネル占有率
- 各ネットワークの MTU

次に、各データ パスのメトリック関数が式 1 に従って計算されます。最初の 3 つの項目がほとんど変わらない点に注意してください。それらの項目は基盤となるネットワーク テクノロジーと相関関係にあり、負荷によって変わることはありません。これらは、コンフィギュレーション ファイルから設定するか、またはオペレータが直接入力することで設定できます。IGRP では、測定遅延は使用されません。理論と実践のどちらの面からも、プロトコルで測定遅延を使用して安定したルーティングを維持するのは非常に難しいことが示唆されています。2 つの測定パラメータがあります:信頼性とチャネル占有率。信頼性は、ネットワーク インターフェイス ハードウェアまたはファームウェアによってレポートされる エラー率に基づいて算出されます。

ルーティング アルゴリズムには、これらの入力値のほかにいくつかのルーティング パラメータが必要です。たとえば、タイマー値、バリアンス、ホールドダウンが有効かどうかなどがそれに該当します。これらのパラメータは、通常、コンフィギュレーション ファイルまたはオペレータの入力によって指定されます (シスコ リリース 8.2 の時点では、バリアンスは常に 1 に設定されています)

初期情報が入力された後は、イベント、つまり、いずれかのネットワーク インターフェイスにデータ パケットが到着するか、またはタイマーが時間切れになることによって、ゲートウェイの動作が起動されます。図 4 ~ 7 のプロセスは次のようにして起動されます。

- パケットが到着すると、図4に従って処理されます。その結果、パケットは別のインターフェイスに送信されるか、廃棄されるか、またはさらなる処理のために受け入れられます。
- さらに処理するためにゲートウェイによって受け入れられたパケットは、プロトコル固有の方法 (この仕様には記述されていない) で解析されます。パケットがルーティング更新の場合は、図 5 に従って処理されます。
- 図 6 は、タイマーによって起動されたイベントを示しています。タイマーは、毎秒 1 回割り込みを生成するように設定されます。割り込みが発生すると、図 6 のプロセスが実行されます。
- 図 7 は、ルーティング更新のサブルーチンを示しています。このサブルーチンのコールは、図 5 と図 6 に含まれています。
- また、図 8 は、図 5 と図 7 で参照されるメトリック計算の詳細を示しています。

経路の伝搬とタイムアウトを制御する重要な時間定数が 4 つあります。これらの時間定数はシステム管理者が設定することもできます。ただし、デフォルト値があります。次にこの時間定数を示します。

- ブロードキャスト時間：この間隔で、すべてのゲートウェイから、接続されたすべてのインターフェイスに関する更新がブロードキャストされます。デフォルトは 90 秒に 1 回です。
- 無効時間：この時間内に特定のパスに関する更新が受信されなかった場合は、タイムアウトしたと見なされます。この値は、更新を含むパケットがネットワークによって廃棄される可能性を考慮するために、ブロードキャスト時間の倍数になります。デフォルトはブロードキャスト時間の 3 倍です。
- ホールド時間：宛先が到達不能になると (または、メトリックがポイズニングの条件を満た

すほど十分に増加すると）、その宛先は「ホールドダウン」状態に移行します。 この状態の間は、ホールド時間が経過するまで同じ宛先に関する新しいパスは受け入れられません。ホールド時間は、ホールドダウン状態がどれくらい継続するかを示します。この値はブロードキャスト時間の倍数になります。デフォルト値はブロードキャスト時間の 3 倍 + 10 秒です。（Disable Holddowns セクションで述べたように、ホールドダウンを無効にすることもできます）

- フラッシュ時間：この時間内に特定の宛先に関する更新が受信されなかった場合は、ルーティング テーブルからそのエントリが削除されます。無効時間とフラッシュ時間の違いに注意してください。無効時間が経過すると、パスがタイムアウトして削除されます。このパス以外に宛先に到達するパスがない場合、この宛先は到達不能になります。しかし、その宛先に対応するデータベース エントリはそのまま残ります。データベース エントリが保持されるのは、ホールドダウンを実行するためです。それに対して、フラッシュ時間が経過するとテーブルからデータベース エントリが削除されます。この値は、無効時間 + ホールドダウン時間よりも少し長くなります。デフォルトはブロードキャスト時間の 7 倍です。

これらの値は、次の主要なデータ構造を前提とします。これらのデータ構造の異なるセットが、ゲートウェイでサポートされているプロトコルごとに保持されます。各プロトコル内部では、データ構造の異なるセットが、サポートされているタイプ オブ サービスごとに保持されます。

システムが認識している宛先ごとに、宛先へのパスのリスト（ヌルの場合もあります）、ホールドダウン有効時間、および最終更新時間が保持されます。最終更新時間は、別のゲートウェイからの更新にこの宛先のパスが含まれていた最後の時間を示します。また、パスごとに更新時間が保持されています。宛先への最後のパスが削除されると、ホールドダウンが無効でない限り（セクション 5.2.1 を参照）、その宛先はホールドダウン状態に移行します。 ホールドダウン有効時間は、ホールドダウンが時間切れになる時間を示します。これがゼロ以外の値である場合は、宛先がホールドダウン状態にあることを示します。また、計算時間を節約するために、宛先ごとに「最適メトリック」を保持することもいい考えです。これは単純に、宛先へのすべてのパスについての複合メトリックの最小値です。

宛先へのパスごとに、パス内のネクストホップのアドレス、使用されるインターフェイス、およびパスの特性を記述するメトリックのベクター（トポロジ上の遅延、帯域幅、信頼性、チャネル占有率など）が保持されます。その他にも、ホップ カウント、MTU、情報の発信元、リモートの複合メトリック、式 1 に従って算出された複合メトリックなどの情報も各パスに対応付けられます。最終更新時間も保持されます。情報の発信元は、そのパスに関する最新の更新がどこから到達したかを示します。実際には、これはネクストホップのアドレスと同じです。最終更新時間は単純に、そのパスに関する最新の更新が到達した時間を示します。これはタイムアウトしたパスを無効にするために使用されます。

IGRP 更新メッセージには 3 つの部分があります。内部、システム（「この自律システム」という意味ですが、内部ではありません）、および外部です。「内部」セクションはサブネットへの経路のためのものです。必ずしもすべてのサブネット情報が含まれているわけではありません。1 つのネットワークのサブネットのみが含まれています。これは、その更新の送信先アドレスに対応するネットワークです。通常、更新は各インターフェイスでブロードキャストされるため、これは単純に、ブロードキャストが送信されるネットワークになります。（IGRP 要求およびポイントツーポイント IGRP に対する応答では別の状況が発生します。）メジャー ネットワーク（つまり、非サブネット）は、特に「外部」フラグが設定されていない限り、更新メッセージの「システム」部分に含まれます。

ネットワークが別のゲートウェイから学習され、その情報が、到達した更新メッセージの「外部」部分に含まれていた場合は、そのネットワークに「外部」フラグが設定されます。シスコの実装では、システム管理者が特定のネットワークを外部として宣言することもできます。外部経路は「デフォルト候補」とも呼ばれます。 これらは、デフォルトとして適切と考えられるゲートウ

ェイに到達またはそれを通過する経路で、宛先への明示的な経路がない場合に使用されます。たとえばラトガーズ大学では、ラトガーズ大学を地域ネットワークに接続するゲートウェイ上で、NSFnet バックボーンへの経路に外部フラグを設定しています。シスコの実装では、デフォルト ルートとしてメトリックの最も小さい外部経路が選択されます。

次の項では、図 4 ~ 8 の特定の部分について詳しく説明します。

パケット ルーティング

図 4 は、入力パケットの処理全体を示しています。これは、単に用語を明確にするためだけに使用されます。これが IP ゲートウェイの動作すべてについての記述ではないことは、明らかです。

このプロセスでは、サポート対象プロトコルのリストと、ゲートウェイの初期化時に入力されたインターフェイスに関する情報を使用します。パケット処理の詳細は、パケットで使用されるプロトコルによって異なります。これは、ステップ A で決定されます。ステップ A は、図 4 の中で唯一、すべてのプロトコルで共有される部分です。プロトコル タイプがわかると、それ以降はそのプロトコル タイプに対応する図 4 の実装が使用されます。詳細なパケット内容は、プロトコルの仕様に基づいて記述されています。プロトコルの仕様には、パケットの宛先を判断する手順、宛先をゲートウェイ自身のアドレスと比較してゲートウェイ自身が宛先であるかどうかを判断する手順、パケットがブロードキャストであるかどうかを判断する手順、および宛先が指定されたネットワークの一部であるかどうかを判断する手順が含まれます。これらの手順は、図 4 のステップ B と C で使用されます。ステップ D のテストでは、ルーティング テーブルにリストされている宛先を検索する必要があります。このテストは、ルーティング テーブル内に宛先のエントリがあり、その宛先に少なくとも 1 つの使用可能なパスが対応付けられている場合に真となります。このステップと次のステップで使用される宛先とパスのデータは、サポートされるタイプ オブ サービスごとに別々に管理されています。したがって、このステップは、パケットで指定されているタイプ オブ サービスを判断し、対応するデータ構造のセットを選択することから始まります。そこで選択されたデータ構造のセットが、このステップと次のステップで使用されます。

パスのリモートの複合メトリックがパス自体の複合メトリックよりも小さい場合、そのパスはステップ D と E の目的で使用できます。リモートの複合メトリックがパス自体の複合メトリックよりも大きいパスは、メトリックを基準として、ネクストホップが宛先よりも遠いパスです。このようなパスを「アップストリーム パス」と呼びます。 通常は、メトリックを使用することで、アップストリーム パスが選択されることは回避されると考えられます。アップストリーム パスが最適パスになり得ないことは容易に理解できます。ただし、大きなバリアンスが許容されている場合は、最適パス以外のパスが使用される可能性があります。それらの中にアップストリームが含まれることがあります。

ステップ E では使用するパスを計算します。リモートの複合メトリックがパス自体の複合メトリックよりも大きいパスは対象外となります。複数のパスが受け入れ可能な場合、それらのパスは重み付けラウンドロビン方式で交互に使用されます。特定のパスが使用される頻度は、そのパスの複合メトリックに反比例します。

ルーティング更新の受信

図 5 は、近接ゲートウェイから受信されたルーティング更新の処理を示しています。ルーティング更新はエントリのリストから構成されており、各エントリは 1 つの宛先に関する情報を含んでいます。複数のタイプ オブ サービスを収容するため、1 つのルーティング更新の中に、同じ宛先に対する複数のエントリが含まれる場合があります。図5に示すように、これらのエントリはそれぞれ個別に処理されます。エントリが更新の外部セクションにある場合、このプロセスの結果として追加されると、宛先に外部フラグが設定されます。

図 5 のプロセス全体を、ゲートウェイでサポートされているタイプ オブ サービスごとに 1 回ずつ、そのタイプ オブ サービスに対応する宛先/パス情報のセットを使用して繰り返す必要があります。これは、図 5 の一番外側のループで示されています。ルーティング更新全体は、タイプ オブ サービスごとに 1 回ずつ処理する必要があります。(IGRP の現在の実装では、複数のタイプ オブ サービスはサポートされていません。したがって、一番外側のループは、実際には実装されていません)。

ステップ A で、パスに関する基本的な受け入れ可能性のテストが実施されます。これには、宛先の妥当性テストが含まれます。ありえない (「Martian」) ネットワーク番号は拒否されます。(詳細はRFC 1009 およびRFC 1122を参照)。 また、更新によって参照されている宛先がホールドダウン状態にある場合、つまり、ホールドダウン有効時間がゼロ以外で、現在の時刻より後の場合にも、更新が拒否されます。

ステップ B でルーティング テーブルが検索され、このエントリが示すパスが既知のものであるかどうかが確認されます。ルーティング テーブル内のパスは、そのパスが対応付けられている宛先、パスの一部としてリストされているネクストホップ、そのパスで使用される出力インターフェイス、および情報の発信元 (更新の発信元アドレス -- 実際には、ほとんどの場合ネクストホップと同じ) によって定義されています。 更新パケット内のエントリに記述されているパスは、エントリ内にその宛先がリストされています。また、更新が到着したインターフェイスがそのパスの出力インターフェイスとなり、更新を送信したゲートウェイのアドレス (「source」 S) がそのパスのネクストホップと情報の発信元になります。

ステップ H と T で、図 7 の更新プロセスがスケジュールされます。このプロセスは、実際には、図 5 のプロセス全体が終了した後に実行されます。つまり、図7に示す更新プロセスは、図5に示す処理中に何度もトリガーされた場合でも、一度だけ実行されます。さらに、ネットワークが急速に変化する場合は、更新が頻繁に発行されないように、予防策を講じる必要があります。

ステップ K は、更新パケット内の現在のエントリに含まれる宛先がルーティング テーブルにすでに存在している場合に実行されます。K では、更新パケット内のデータから算出された新しい複合メトリックと、宛先に対する最適な複合メトリックを比較します。このときは、最適な複合メトリックは再計算されません。そのため、対象のパスがルーティング テーブルにすでに存在する場合、このテストは同じパスに対する新旧のメトリックの比較になることがあります。

ステップ L は、既存の最適な複合メトリックよりもメトリックが大きいパスに対して実行されます。これには、既存のパスよりもメトリックが大きい新しいパスと、複合メトリックが増加した既存のパスの両方が含まれます。ステップ L では、新しいパスが受け入れ可能かどうかをテストします。ここでは、新しいパスが保持するための条件を満たすほど十分良好であるかを判断するテストと、ルート ポイズニングのテストの両方が実行されます。パスを受け入れるには、遅延値が到達不能の宛先を示す特別な値 (現在の IP 実装では、24 ビット フィールドがすべて 1) でなく、複合メトリック (図 8 に従って計算された値) が受け入れ可能であることが必要です。複合メトリックが受け入れ可能であるかどうかを判断するには、それを宛先への他のすべてのパスの複合メトリックと比較します。M をこれらの中で最小の値とします。新しいパスは、そのメトリックが V x M よりも小さい場合に受け入れられます。V は、ゲートウェイの初期化時に設定されたバリアンスです。V = 1 の場合 (これは、シスコ リリース 8.2 の時点では常に真です)、既存のメトリックよりも大きいメトリックは受け入れられません。これには例外が ひとつあります。パスがすでに存在し、なおかつそれが宛先への唯一のパスである場合に、メトリックの増加率が 10 % を超えていなければ (あるいはホールドダウンが無効の場合はホップ カウントが増加していなければ)、そのパスは保持されます。

ステップ V は、パスの新しい情報が、複合メトリックが減少したことを示す場合に実行されます。宛先 D へのすべてのパスの複合メトリックが比較されます。この比較では、ルーティング テーブルに保持されているメトリックではなく、P の新しい複合メトリックが使用されます。複合メトリックの最小値 M が算出されます。続いて、D へのすべてのパスが再び検査されます。複合メ

トリックが M x V よりも大きいパスがあれば、そのパスは削除されます。V はバリアンスで、ゲートウェイの初期化時に入力されます (シスコ リリース 8.2 の時点では、バリアンスは常に 1 に設定されています)

定期的処理

図 6 のプロセスは毎秒 1 回起動されます。このプロセスでは、ルーティング テーブル内の各種タイマーが時間切れになっていないかどうかを検査します。これらのタイマーについてはすでに説明しています。

ステップ U で、図 7 のプロセスが起動されます。

ステップ R と S が必要となるのは、ルーティング テーブルに保持されている複合メトリックはチャネル占有率に左右され、チャネル占有率は測定に基づいて経時的に変化するためです。チャネル占有率は、インターフェイスを通過するトラフィックの測定値の移動平均を使用して、定期的に再計算されます。新しく算出された値が既存の値と異なる場合は、そのインターフェイスに関係するすべての複合メトリックを調整する必要があります。ルーティング テーブル内のすべてのパスが検査されます。そのネクストホップがインターフェイス「I」を使用しているパスはすべて、複合メトリックが再計算されます。この計算は式 1 に従って行われますが、チャネル占有率には、パスのメトリックの一部としてルーティング テーブルに保持されている値の最大値と、そのインターフェイスの新しく算出されたチャネル占有率が使用されます。

更新メッセージの生成

図 7 は、ゲートウェイが他のゲートウェイに送信する更新メッセージをどのようにして生成するかを示しています。ゲートウェイに接続されたネットワーク インターフェイスごとに異なるメッセージが生成されます。生成されたメッセージは、各インターフェイスから到達可能な他のすべてのゲートウェイに対して送信されます (ステップ J)。 通常、このメッセージはブロードキャストとして送信されます。ただし、ネットワーク テクノロジーまたはプロトコルでブロードキャストが許可されていない場合は、各ゲートウェイに個別にメッセージを送信する必要があります。

一般に、このメッセージの作成時には、ルーティング テーブル内の宛先ごとにエントリが追加されます (ステップ G)。各タイプ オブ サービスに対応する宛先/パス データを使用する必要があります。最悪のケースでは、各タイプ オブ サービスの宛先ごとに、新しいエントリが更新に追加されます。ただし、ステップ G で更新メッセージにエントリを追加する前に、すでに追加されているエントリがスキャンされます。新しいエントリが更新メッセージ中にすでに存在している場合は、もう一度追加されることはありません。新しいエントリは、宛先とネクストホップ ゲートウェイが同じ場合に既存のエントリをコピーします。

簡単にするために、疑似コードには次の3つの部分があります。IGRP更新メッセージには次の3つの部分があります。その 3 つとは、内部、システム、および外部であり、したがって、実際には宛先全体にわたる 3 つのループが存在します。1 つめのループには、更新の送信先ネットワークのサブネットのみが含まれます。2 つめのループには、外部フラグが設定されていないすべてのメジャー ネットワーク (つまり、非サブネット) が含まれます。3 つめのループには、外部フラグが設定されているすべてのメジャー ネットワークが含まれます。

ステップ E では、スプリット ホライズン テストを実行します。通常、更新が送出されるインターフェイスと同じインターフェイスを最適パスが通過する経路では、このテストは失敗します。しかし、更新が特定の宛先に送信される場合 (たとえば、別のゲートウェイからの IGRP 要求への応答、または「ポイントツーポイント IGRP」の一部として送信される場合など) は、最適パスが元々その宛先から到達したもの (「情報の発信元」が宛先と同じ) で、なおかつ、その要求

が到達したインターフェイスと最適パスの出力インターフェイスが同じ場合に限り、スプリットホライズンは失敗します。

メトリック情報の計算

図 8 は、ゲートウェイで受信された更新メッセージからメトリック情報がどのようにして処理され、さらにゲートウェイから送信される更新メッセージでメトリック情報がどのようにして生成されるかを示しています。エントリは、宛先への ひとつの特定のパスに基づきます。宛先へのパスが複数ある場合は、複合メトリックが最小のパスが選択されます。複合メトリックに最小のパスが複数ある場合は、任意の同点決戦規則が使用されます (ほとんどのプロトコルで、これはネクストホップ ゲートウェイのアドレスに基づきます)

図 4 : 着信パケットの処理

```
Data packet arrives using interface I

A   Determine protocol used by packet

    If protocol is not supported
      then discard packet

B   If destination address matches any of gateway's addresses
    or the broadcast address
      then process packet in protocol-specific way

C   If destination is on a directly-connected network
      then send packet direct to the destination, using
      the encapsulation appropriate to the protocol and link type

D   If there are no paths to the destination in the routing
    table, or all paths are upstream
      then send protocol-specific error message and discard the packet

E   Choose the next path to use. If there are more than
    one, alternate round-robin with frequency proportional
    to inverse of composite metric.

    Get next hop from path chosen in previous step.

    Send packet to next hop, using encapsulation appropriate
    to protocol and data link type.
```

図 5 : 着信ルーティング アップデートの処理

```
Routing update arrives from source S

    For each type of service supported by gateway
      Use routing data associated with this type of service

      For each destination D shown in update

A       If D is unacceptable or in holddown
          then ignore this entry and continue loop with next destination D

B       Compute metrics for path P to D via S (see Fig 8)

        If destination D is not already in the routing table
        then Begin
```

```
          Add path P to the routing table, setting last
          update times for P and D to current time.

H         Trigger an update

          Set composite metric for D and P to new composite
          metric computed in step B.

          End

        Else begin (dest. D is already in routing table)

K         Compare the new composite metric for P with best
          existing metric for D.

           New > old:

L            If D is shown as unreachable in the update,
             or holddowns are enabled and
              the new composite metric >
                (the existing metric for D) * V
                [use 1.1 instead of V if V = 1,
                as it is as of Cisco release 8.2]
O            or holddowns are disabled and
             P has a new hop count > old hop count
             then Begin

              Remove P from routing table if present

              If P was the last route to D
                then Unless holddowns are disabled
                    Set holddown time for D to
                     current time + holddown time
T                  and Trigger an update

              End

             else Begin

              Compute new best composite metric for D

              Put the new metric information into the
              entry for P in the routing table

              Add path P to the routing table if it
              was not present.

              Set last update times for P and D to
              current time.

              End

           New <= OLD:

V            Set composite metric for D and P to new
             composite metric computed in step B.

             If any other paths to D are now outside the
             variance, remove them.

             Put the new metric information into the
             entry for P in the routing table
```

```
              Set last update times for P and D to
              current time.

            End

          End of for

        End of for
```

図 6：定期的処理

```
Process is activated by regular clock, e.g. once per second

    For each path P in the routing table (except directly
    connected interfaces)

      If current time < P'S LAST UPDATE TIME + INVALID TIME
        THEN CONTINUE WITH THE NEXT PATH P

      Remove P from routing table

      If P was the last route to D
        then Set metric for D to inaccessible
          Unless holddowns are disabled,
            Start holddown timer for D and
          Trigger an update

        else Recompute the best metric for D

      End of for

    For each destination D in the routing table

      If D's metric is inaccessible
        then Begin

        Clear all paths to D

        If current time >= D's last update time + flush time
          then Remove entry for D

        End

      End of for

    For each network interface I attached to the gateway

R     Recompute channel occupancy and error rate

S     If channel occupancy or error rate has changed,
        then recompute metrics

      End of for

    At intervals of broadcast time

U     Trigger update
```

図 7：更新の生成

```
Process is caused by "trigger update"
```

```
    For each network interface I attached to the gateway

      Create empty update message

      For each type of service S supported

        Use path/destination data for S

        For each destination D

E         If any paths to D have a next hop reached through I
            then continue with the next destination

          If any paths to D with minimal composite metric are
          already in the update message
            then continue with the next destination

G         Create an entry for D in the update message, using
          metric information from a path with minimal
          composite metric (see Fig. 8)

          End of for

        End of for

J     If there are any entries in the update message
       then send it out interface I

      End of for
```

図 8：メトリック計算の詳細

この項では、受信されたルーティング更新からメトリックとホップ カウントを計算する手順について説明します。この関数への入力は、ルーティング更新パケットに含まれる特定の宛先の工ントリです。出力は、複合メトリックの計算に使用できるメトリックのベクターとホップ カウントです。このパスがルーティング テーブルに追加される場合は、メトリックのベクター全体がテーブルに入力されます。次の定義で使用されるインターフェイス パラメータは、ゲートウエイの初期化時に設定されたパラメータのうち、ルーティング更新が到達したインターフェイスに関するものです。ただし、チャネル占有率と信頼性は除きます。これらは、インターフェイスを通過するトラフィックの測定値の移動平均に基づきます。

- 遅延 = パケットに含まれる遅延 + インターフェイスのトポロジ上の遅延
- 帯域幅 = max (パケットに含まれる帯域幅、インターフェイスの帯域幅)
- 信頼性 = min (パケットに含まれる信頼性、インターフェイスの信頼性)
- チャネル占有率 = max (パケットに含まれるチャネル占有率、インターフェイスのチャネル占有率) (帯域幅で max が使用されるのは、帯域幅メトリックが逆数で保存されているためです。概念的には、最小の帯域幅を使用する必要があります。) パケットに含まれる元のチャネル占有率は保存しておく必要があります。これは、インターフェイスのチャネル占有率が変化するたびにチャネル占有率の実効値を再計算する必要があるためです。

次のパラメータはメトリック ベクターのコンポーネントではありませんが、これらもパスの特性としてルーティング テーブルに保持されます。

- ホップ カウント = パケットに含まれるホップ カウント
- MTU = min (パケットに含まれる MTU、インターフェイスの MTU)
- リモートの複合メトリック -- パケットに含まれるメトリック値を使用して、式 1 から算出されます。つまり、上記のように更新されたメトリックではなく、パケットに含まれているメトリックがメトリック コンポーネントになります。明らかにこれは、上記の調整を行う前に

計算する必要があります。
- 複合メトリック -- この項の説明に従って計算されたメトリック値を使用して、式 1 から算出されます。

この項では、送信されるルーティング更新用のメトリックとホップ カウントを計算する手順について説明します。

この関数は、発信更新パケットに収められるメトリック情報とホップ カウントを決定します。使用可能なパスがある場合、これは宛先への特定のパスに基づきます。パスがない場合、またはパスがすべてアップストリームの場合、その宛先は「アクセス不能」と呼ばれます。

```
If destination is inaccessible, this is indicated by using a specific
        value in the delay field. This value is chosen to be larger
        than the largest valid delay. For the IP implementation this is
        all ones in a 24-bit field.

   If destination is directly reachable through one of the interfaces, use
        the delay, bandwidth, reliability, and channel occupancy of the
        interface. Set hop count to 0.

   Otherwise, use the vector of metrics associated with the path in the
        routing table. Add one to the hop count from the path in the
        routing table.
```

IP 実装の詳細

この項では、シスコの IGRP 実装で使用されているパケットのフォーマットについて簡単に説明します。IGRP は、IP プロトコル 9 (IGP) を含む IP データグラムを使用して送信されます。 パケットはヘッダーで始まります。このヘッダーは、IP ヘッダーのすぐ後から始まります。

```
unsigned version: 4;  /* protocol version number */
    unsigned opcode: 4;   /* opcode */
    uchar edition;        /* edition number */
    ushort asystem;       /* autonomous system number */
    ushort ninterior;     /* number of subnets in local net */
    ushort nsystem;       /* number of networks in AS */
    ushort nexterior;     /* number of networks outside AS */
    ushort checksum;      /* checksum of IGRP header and data */
```

更新メッセージの場合、ヘッダーの直後にルーティング情報が続きます。

バージョン番号は、現時点では 1 です。他のバージョン番号を持つパケットは無視されます。

opcode は次のいずれかです。1 更新 2 要求

これはメッセージのタイプを示します。これら 2 つのメッセージ タイプのフォーマットは後述します。

edition は、ルーティング テーブルに変更があるたびに増加するシリアル番号です。(上の疑似コードがルーティング アップデートをトリガーする条件下で実行されます)。 ゲートウェイでは、エディション番号を使用して、すでに処理済みの情報を含む更新を再び処理することを回避できます。(これは、現在は実装されていません。つまり、エディション番号は正しく生成されますが、入力時には無視されます。パケットは廃棄される可能性があるため、エディション番号が処理の重複を回避するために十分有効であると確信することはできません。エディション番号を使用するのであれば、エディションと対応付けられたすべてのパケットが確実に処理されるようにする必要があります)。

*asystem は自律システム番号です。*シスコの実装では、1 台のゲートウェイが複数の自律システムに参加できます。自律システムはそれぞれ独自に IGRP プロトコルを実行します。概念的には、自律システムごとにまったく異なるルーティング テーブルがあります。IGRP を通じて ひとつの自律システムから到達する経路は、その AS 用の更新のみに含まれて送信されます。ゲートウェイでは、このフィールドに基づいて、メッセージの処理に使用するルーティング テーブルのセットを選択できます。ゲートウェイが未設定の AS に関する IGRP メッセージを受信した場合、そのメッセージは無視されます。実際は、シスコの実装では AS 間で情報を「漏洩」できるようになっています。ただし、筆者はこれをプロトコルの一部ではなく、管理ツールと見なしています。

*ninterior、nsystem、および nexterior は、それぞれ更新メッセージの 3 つのセクションに含まれるエントリの数を示しています。*これらのセクションについてはすでに説明しています。これらのセクションの間には区切りがありません。最初から ninterior 個のエントリが内部と見なされ、次の nsystem 個のエントリがシステム、最後の nexterior 個のエントリが外部と見なされます。

checksum は IP チェックサムで、UDP チェックサムと同じチェックサム アルゴリズムを使用して計算されます。チェックサムは、IGRP ヘッダーとそれに続くルーティング情報について計算されます。checksum フィールドは、チェックサムを計算するときにゼロに設定されます。チェックサムは IP ヘッダーを含みません。また、UDP と TCP のような仮想ヘッダーもありません。

要求

IGRP 要求は、受信側ゲートウェイに対してルーティング テーブルを送信するよう要求します。要求メッセージにはヘッダーしかありません。使用されるのは version、opcode、および asystem フィールドのみです。その他のフィールドはすべてゼロに設定されます。受信側ゲートウェイは、要求側に対して通常の IGRP 更新メッセージを送信するように求められます。

アップデート

IGRP 更新メッセージにはヘッダーがあり、その直後からルーティング エントリが続きます。1 つの更新メッセージには、1500 バイトのデータグラム (IP ヘッダーを含む) に収まるだけのルーティング エントリが含まれます。 現在の構造宣言では、最大で 104 エントリを収容できます。それよりも多くのエントリが必要とされる場合は、複数の更新メッセージが送信されます。更新メッセージは単純にエントリ別に処理されるため、複数の独立したメッセージを使用する代わりに単一フラグメントのメッセージを使用しても何もメリットはありません。

ルーティング エントリの構造を次に示します。

```
uchar number[3];       /* 3 significant octets of IP address */
    uchar delay[3];        /* delay, in tens of microseconds */
    uchar bandwidth[3];    /* bandwidth, in units of 1 Kbit/sec */
    uchar mtu[2];          /* MTU, in octets */
    uchar reliability;     /* percent packets successfully tx/rx */
    uchar load;            /* percent of channel occupied */
    uchar hopcount;        /* hop count */
```

uchar[2] と uchar[3] で定義されたフィールドは、通常の IP ネットワークのレベルでは単純に 16 ビットおよび 24 ビットの 2 進整数です。

number は記述される宛先を定義します。これは IP アドレスです。スペースを節約するために、内部セクションを除き、IP アドレスの最初の 3 バイトが指定されます。内部セクションでは最後の 3 バイトが指定されます。システム経路および外部経路では、サブネットである可能性はないため、下位バイトは常にゼロになります。内部経路は常に既知のネットワークのサブネットであ

るため、そのネットワーク番号の最初のバイトが指定されます。

delay の単位は 10 マイクロ秒です。このフィールドには、10 マイクロ秒から 168 秒までの範囲内で十分と思われる値が指定されます。delay フィールドがすべて 1 の場合は、そのネットワークに到達できないことを示します。

帯域幅は、1.0e10倍のビット/秒の逆帯域幅です。範囲は1200 bpsの回線から10 Gbpsです。（つまり、帯域幅が N Kbps の場合、使用される数値は 10000000 / N になります）

mtu の単位はバイトです。

信頼性は255の割合として与えられます。つまり、255は100%です。

load は 255 の割合として指定されます。

hopcount は単純なカウントです。

帯域幅と遅延で使用される単位は見慣れないものなので、例を示した方がいいでしょう。いくつかの一般的なメディアで使用されるデフォルト値は次のとおりです。

	Delay	Bandwidth
Satellite	200,000 (2 sec)	20 (500 Mbit)
Ethernet	100 (1 ms)	1,000
1.544 Mbit	2000 (20 ms)	6,476
64 Kbit	2000	156,250
56 Kbit	2000	178,571
10 Kbit	2000	1,000,000
1 Kbit	2000	10,000,000

メトリックの計算

シスコ バージョン 8.0(3) における複合メトリックの実際の計算方法を次に示します。

```
metric = [K1*bandwidth + (K2*bandwidth)/(256 - load) + K3*delay] *
                        [K5/(reliability + K4)]

    If K5 == 0, the reliability term is not included.

    The default version of IGRP has K1 == K3 == 1, K2 == K4 == K5 == 0
```

関連情報

- IP ルーティングに関するサポート ページ
- IGRP サポート ページ
- テクニカルサポート - Cisco Systems