

マルチホーム環境におけるアプリケーションルーティング技術の提案

中川 郁夫	上谷 一	鍋島 公章
インテックシステム研究所	東北大学	NTT ソフトウェア研究所
樋地 正浩		今野 幸典
東北インターネット協議会		東北インターネット協議会

概要

近年、インターネットサービスプロバイダはもちろん、大学や企業などにおいても、複数箇所でのインターネット接続する「マルチホーム」環境を構築している組織が増えてきている。マルチホームは、冗長性の向上や負荷分散に有効な技術とされているが、同時に経路制御が非常に困難であることも指摘されている。

これまでのインターネット技術では、複数の経路が存在する場合にどの経路でトラフィックを流すかについてアプリケーション側で制御することはなかった。このため、マルチホーム環境では必ずしも効率的なトラフィック制御はできないのが実情である。

本稿では、Web (World Wide Web) のキャッシュサーバや代理サーバ技術を応用し、これらのサーバ上で経路情報などの情報を利用することにより、アプリケーション層でのトラフィック制御を行う手法を提案する。本稿で提案される技術を用いることによって、マルチホームを行う組織はネットワーク層の経路制御手法に依存することなく、容易にトラフィック分散が可能になる。

Application routing technology for multi-homed environment

Ikuo Nakagawa	Hajime Kamiya	Masaaki NABESHIMA
INTEC Systems Laboratory	Tohoku University	NTT Software Laboratories
Masahiro Hiji		Yukinori Konno
Tohoku internet Association		Tohoku internet Association

Abstract

Recently, not only internet service providers but also universities or corporations are in multi-homed environment, that is, they are connected to the Internet at multiple points. Multi-homed sites have benefits of redundancy or load balancing. But it is also pointed out that it's very hard to control or manage routing on these environment.

With current internet technologies, most applications do not care about which path should be used to send or receive data traffic when multiple path exist for a destination.

In this paper, we propose application routing technology which controls application data traffic by using external information such as routing table. This is an extension of Web (World Wide Web) proxy technology. Using this new technology suggested in this paper, one can easily control application data traffic, independently to routing management.

1 はじめに

近年、インターネットに接続する組織では、冗長性や負荷分散を目的としてインターネットへの接続点を複数もつ「マルチホーム環境」を構築するケースが増えている。もともとインターネットはベストフォート型の通信を基本としているが、インターネット接続を行っている組織では、障害や遅延などの問題が表面化しており、何らかの対応が必要になってきている。そのため、プロバイダはもちろん、大学や企業であっても、みずから冗長性の確保や負荷分散による効率化を行うケースが増えているものと考えられる。

しかし、マルチホーム環境における経路制御はネットワークポロジや利用するアドレス空間、経路制御のポリシーなどが影響し、その設計・運用には非常に高度な技術を必要とする。特にネットワーク層での経路制御だけで効率的なトラフィック分散を実現しようとする場合、外部へ広告する経路や、受け取った経路を個別に操作することも必要となり、運用コストは膨大なものとなる。

これまでのインターネット技術では、ネットワーク層での経路制御によりトラフィック制御を行うことが多かった。すなわち、複数存在する経路のうち、どの経路を用いてトラフィックを流すかについてアプリケーション側で制御することはなかった。このため、ネットワーク層による経路制御だけでトラフィックの効率化を実現しようとした場合、上記のような経路制御上の問題が表面化し、マルチホーム環境の運用は困難になっているようである。

本稿ではアプリケーション層において複数の経路を使い分けることを提案する。すなわち、アプリケーションにトラフィックを有効に分散させるための仕組みを実装することにより、ネットワーク層での経路制御に依存することなく、負荷分散、冗長性の確保を実現する。

特に、本稿ではインターネット全体のトラフィックの80~90%を占めると言われている World Wide Web (以下 Web) のアクセスについてトラフィック制御を行なう事を検討する。本来 Web のトラフィックは単純なクライアント・サーバ間通信であるが、最近ではトラフィックの軽減やセキュリティ上の理由などからキャッシュサーバや代理サーバを利用するケースが増えて来ている。マルチホーム環境の組織でも、これらの技術を応用することでネットワーク層の経路制御に依存せずにトラフィックを制御することが可能になる。本稿では、これらの代理サーバにおいて経路情報などの外部情報を利用するこ

とにより、効率的に Web のトラフィックを制御する手法を提案する。

本稿では、第2章でマルチホーム環境における経路制御の問題について簡単にまとめる。次に第3章で代理サーバで経路情報などの外部情報を利用する手法について提案し、第3.2節および第3.3節では本手法の実現方針と、経路情報を用いた実現方法とについて言及する。

2 マルチホーム環境における経路制御の問題

マルチホーム環境の組織では、いくつかの要因によって経路制御の実現が複雑になると言われている。本節ではいくつかの典型的なケースにおいて、どのように経路制御を行う必要があるのかについて検討し、その問題について説明する。

2.1 大きなアドレス空間を有している場合

インターネットでは、古くからインターネットに接続を行っている組織には大きなアドレスブロックが割り当てられてきた。特に大学や大手企業などではクラスBのアドレス空間が割り当てられているケースも多い。このように大きなアドレス空間を割り当てられている組織がマルチホーム環境を構築する場合、経路制御はある程度単純なものにすることができる。具体的には組織 A がアドレス空間 Aa を有しており、二つのプロバイダ X, Y に接続をしている場合、組織 A が X および Y と BGP4[1] による経路情報の交換を行うことによりもっとも効率的な経路制御が可能になる。

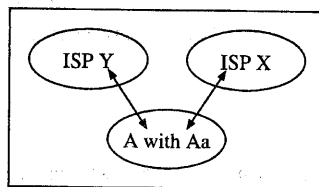


図 1: BGP4 による経路制御

なお、BGP4 の運用には AS (Autonomous System) 番号が必要になるが、AS 番号の数は有限であることからむやみに AS 番号を取得することはインターネットリ

ソースの浪費につながるとされている。そのため、最近では AS 番号を取得する前にじゅうぶんな検討が必要である。なお、組織 A が X や Y と BGP4 による経路情報の交換を行うにはその組織間だけで有効なプライベート AS 番号を使うこともある。この場合、A は X および Y から経路情報を取得し、A の経路情報の広告は (A ではなく) X および Y が代理で行うことになる。

以上のように大きなアドレス空間を持つ組織の場合、BGP4 による経路制御により比較的単純な経路制御が可能であるが、実際には次のような問題も抱えている。

- 世界中の経路情報 (full route) は現在 60,000 程度あり、これらの全てを X および Y から受け取るためには、ある程度の回線の帯域が必要であり、さらにその組織、およびプロバイダの双方に高性能なルータが必要になる。
- BGP4 による経路の頻繁な更新 (route flap) などに対応した経路制御が必要になる。
- インターネットへ接続しているルータが複数ある場合、それらのルータを含め、組織内のルータの一部で上記の全経路情報を交換する必要がある。
- BGP4 の運用を行うには、経路制御技術についての詳しい知識が必要であり、一般の組織では運用が重荷になることが多い。

実際には、世界中の経路情報を受け取ることは組織のみならずプロバイダにも負担がかかるため、プロバイダからは一部の経路のみを伝搬してもらい、それ以外はデフォルトの指定により運用していることも多い。

なお、最近ではプロバイダ毎の CIDR ブロックに従ったアドレス割り当てが行われており、上記のような大きなアドレス空間の割り当てを受けることは非常に難しくなっている。今後、一般の組織では大きなアドレス空間の割り当てを受けることはほぼ不可能であり、上記のようなケースは非常にまれなケースとなるだろう。

2.2 複数のプロバイダから個別にアドレスを割り当てられる場合

インターネットに接続された組織のうち、比較的新しい組織などではプロバイダから接続点アドレスと呼ばれる小さいアドレス空間を割り当てられているのが一般的である。例えば、ある組織 A がふたつのプロバイダ X

と Y に接続を行っている場合に X から Xa というアドレス空間を、Y から Ya というアドレス空間をそれぞれ割り当てられているものとする。

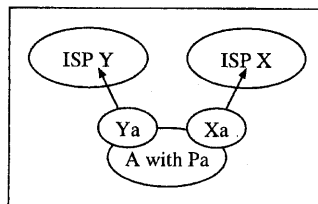


図 2: 接続点アドレスによる接続

この場合、組織 A の中には Xa のアドレス空間を持つ部分と Ya のアドレス空間を持つ部分が共存することになる。最近では、こうしたアドレス空間の割り当てはプレフィクス長が 24~29 の単位で行われており、それぞれから割り当てられたアドレス空間は非常に小さいものになる。

本ケースのような小さなアドレス空間についての個別の経路情報をインターネットに対して広告するのは、経路情報の爆発などを引き起こすことになるため一般には行なわれていない。特に最近では、米国の大手プロバイダなどでプレフィクス長が 19 を越えるような小さなアドレス空間については経路情報の交換がされないケースもある。仮に経路を広告してもインターネット上での到達性は保証されなくなってきたということになる。

また、近年問題視されている「なりすまし」等に対するセキュリティ上の理由から、Xa のアドレスを持つ IP パケットを Y 経由で流すこともフィルタにより抑制されているケースが増えている。

以上の理由から、Xa のアドレスを持つホストは X 経由で、Ya のアドレスを持つホストは Y 経由でインターネット上のホストと通信を行うのが一般的である。この場合、プライベート AS 番号の利用により、全経路を受け取ることは可能であるが、宛先アドレスによる経路の選択以前に、ソース (発信元) アドレスによって経路を決定する必要があるため、経路表を有効に利用することはできない。

このようなケースでは、それぞれのホストやルータについてはほぼ静的な経路制御により運用されることも多いが、実際に組織内のネットワークと連係して経路制御を実現しようとする場合には、以下のような問題がある。

- Xa のホストと Ya のホスト間での通信を内部で行なうため、Xa と Ya 上のルータおよびホストは、それぞれ他方の経路情報を管理する必要がある。
- ひとつのホストで Xa および Ya の両方のアドレスを持つ場合、このホストでは、Xa のアドレスを持つ IP パケットは X 経由で、Ya のアドレスを持つ IP パケットは Y 経由で通信する必要がある。すなわち、ソースアドレスルーティングの技術を用いなければならない。
- 宛先で経路を決定できないため、ネットワーク層での経路の最適化は非常に困難である。

3 アプリケーションルーティング技術

前節で述べた通り、アドレス空間の枯渇問題や経路情報の爆発問題などを考慮すると、今後は第 2.2 節で述べた、プロバイダごとに「接続点アドレス」を割り当ててもらおう方法でのマルチホーム環境が増えてくることが予想される。

このようなケースでは、トラフィックの負荷分散を行う際にネットワーク層で最適経路の選択を行うことは難しいため、アプリケーション層での工夫 [3][5][2] が必要である。

本節では、インターネットのトラフィックのうち 80~90% を占めると言われている Web のアクセスを例に、アプリケーション層での最適経路の選択を行なう方法(アプリケーションルーティング)について検討する。特に、最近では Web のアクセスは代理サーバを経由して行われることが多いため、本稿でもこれらの技術を応用することにより効率的なトラフィック分散を実現することを目指す。

代理サーバ間の協調動作のひとつである親子関係 [4] は、アプリケーション層によるルーティングのひとつであると言える。現状では、これはドメイン名による比較や固定的な AS 番号の比較であるため、実際のネットワーク層でのルーティングとは異なり、必ずしも最適な経路を選択できるとは限らない。本稿で提案する手法は、これらの問題を解決し、アプリケーション層において最適な経路を選択するためのフレームワークになる。

なお、本稿では複数の接続点アドレスによるマルチホーム環境を例にアプリケーション層の経路選択手法について提案するが、本稿に示される技術は他のマルチ

ホーム技術についても容易に応用が可能である。

3.1 アプリケーションルーティングの仕組み

マルチホーム環境にある組織 A が、接続を行なっているプロバイダ X および Y からそれぞれ個別に Xa, Ya という接続点アドレスを割り当てられているものとする。一般に Xa, Ya は非常に小さいアドレス空間であることが多く、このような組織では内部でプライベートアドレス空間を利用することが多い。ここでは、組織 A がプライベートアドレス Pa により内部ネットワークを構築しているものとする。

組織 A のユーザが Web のアクセスを行なう場合、ユーザは外部サブネットに設置された代理サーバを経由してインターネット上のサーバにアクセスを行なうが、ここでは次のような構成を考える。

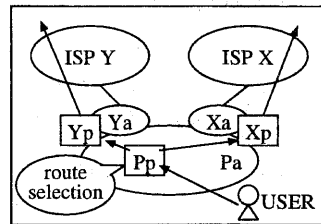


図 3: アプリケーションルーティング

Xp および Yp は、それぞれ Xa, Ya のサブネットに設置された代理サーバを表している。

ユーザは、一旦 Pa のアドレスを持つ首振りサーバ Pp に Web アクセスの要求を行なう。首振りサーバは、受け取った Web アクセスについて最適な経路を計算し Xa および Ya に設置された代理サーバに要求を転送する。Xa および Ya に設置された代理サーバ Xp, Yp は、それぞれが接続するプロバイダを経由して目的のサーバへアクセスを行ない、特に経路を計算しないものとする。

この仕組みにおいて、実際にアプリケーション層で経路選択を行なうのは Pa のアドレスを持つ首振りサーバである。首振りサーバは何らかの外部情報を用いて、組織内部からの Web のアクセス要求を、最適な経路を通るように Xa および Ya に振り分ける必要がある。ここで用いられる外部情報には、次のような情報を利用することが考えられる。

- Xp および Yp から目的のサーバまでの RTT (Round Trip Time)
- Xp および Yp を経由して目的のサーバへアクセスした過去の平均転送時間 [6][7]
- Xp および Yp から目的のサーバまでのルータの数
- 経路情報による距離
- ポリシに依存した静的な情報

次節以降では、これらの外部情報を利用した首振りサーバの実装手段、およびその外部情報として経路情報を利用する場合の実装例について紹介する。

3.2 アプリケーションルーティングの実装方針

本節では、前節までに述べたアプリケーションルーティング技術の実装方針について述べる。著者らは現在アプリケーションルーティング技術を実装し、大規模な Web アクセスが発生する環境での実験を行なうことを予定している。これらの実験では首振りサーバの実装として Squid を利用することを考えている。Squid はフリーの代理サーバとしてもっとも多く利用されており、ソースファイルも公開されている。

Squid はクライアントから Web アクセス要求を受け取った場合、以下のような判断を行ない、次の処理を決定する。なお、本稿での例にある首振りサーバは内部アドレス Pa が割り当てられているため、最後の処理は行なわないことになる。

1. Squid のキャッシュ領域に含まれていればそれを返す
2. sibling に指定された近隣のサーバ上にキャッシュがあればそれを返す
3. 別の代理サーバにリクエストを転送すべきであれば転送処理を行なう
4. Squid みずから目的のホストへアクセスを行なう

ここで、アプリケーションルーティングを行なうには、上記のうち 3 の処理中、転送すべき代理サーバの決定アルゴリズムを変更することになる。具体的には、転送すべき代理サーバを次のような手順で決定することになる。

1. 静的なルールに記述された代理サーバ

2. 外部情報を用いて選択したより適切な代理サーバ
3. デフォルトの代理サーバ

また、本手法は首振りサーバへの変更のみで実現されるため、Xp, Yp などの代理サーバ、および一般のユーザは従来利用していた設定を変更する必要はなく、最小限の変更で本提案による効果を得ることができる。

3.3 アプリケーションルーティングにおける経路情報の利用

前節では首振りサーバで外部情報を利用して経路選択を行なう場合の首振りサーバの実装方針について記述したが、本節では、その外部情報として経路情報を利用する手段について紹介する。

首振りサーバが経路情報を用いて経路選択を行なう場合、同サーバはアプリケーション層から経路情報を取得することができなければならない。また、適切な経路選択を行なうためには、プロバイダ X, Y からそれぞれの経路情報を受け取らなければならない。ただし、ここで受け取る経路情報はネットワーク層で用いる経路情報とは異なり、アプリケーション的に経路選択を行なうためのものである。プロバイダ側の運用を考慮すると、情報受け取るためのプロトコルとしては BGP4 で運用することが自然であるが、必要な経路情報は、プロバイダ X, Y およびその近隣 AS 程度でもじゅうぶん実用的であるといえる。

なお、本稿では首振りサーバを内部アドレス Pa が割り当てられているとしているため、プロバイダ X, Y のルータなどとは直接通信を行なうことはできない。そのため、実際の経路情報の取得に際しては何らかの方法で間接的に経路情報を取得する仕組みが必要である。例えば、プロバイダ X, Y から経路情報を伝搬するためのルータを Xa, Ya それぞれに設置する方法 (図 4) や、経路情報伝搬ようの代理サーバの利用、NAT (Network Address Translation) の利用などが考えられる。もちろん、ネットワーク層の経路制御に BGP4 などを運用している場合にはそれをそのまま利用することも可能である。

首振りサーバでは、クライアントからのリクエストを受けるたびに前節に記述された手順に従って、処理を行なうが、代理サーバを決定する処理が行なわれた場合には、これらの経路情報サーバに対して問い合わせを行ない、宛先ホストまでの距離を計算する。最終的に宛先ホ

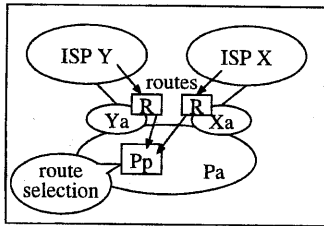


図 4: 首振りサーバにおける経路情報の取得

ストまでの距離が近くなるように、リクエストを転送すべき代理サーバを X_p, Y_p から選択することになる。

4 まとめ

本稿では、マルチホーム環境を持つ組織が効率的なトラフィック制御を行なうために、アプリケーション層で経路制御を行なうアプリケーションルーティング技術について提案した。本稿では、特に Web のトラフィックについて代理サーバの技術を応用し、経路情報などの外部情報を用いてアクセスリクエストを転送する先を決めることによりアプリケーション層での経路選択を可能にする手法について述べた。

本稿では Web トラフィックに限定してその実装手段について述べたが、アプリケーションルーティング技術はインターネットの多くのアプリケーションで応用が可能で、これらの技術を組みあわせることによって、ネットワーク層に依存せず、トラフィック交換のための効率的な経路選択が可能になる。

なお、著者らは本稿に述べた Web に関するアプリケーションルーティング技術について実装し、APAN および国内の地域 IX などでも実験を行なっていく予定である。

謝辞:

本研究にご協力いただいた、東北地域内インターネット相互接続実験研究会のみなさま、と APAN のみなさまに感謝します。

参考文献

- [1] Y. Rekhter, T. Li: "A Border Gateway Protocol 4", RFC1771, Mar. 1995
- [2] 菊池豊, 菊地時夫
"PIX:応用層によるトラフィック交換モデル", インターネットコンファレンス'97, Dec 1997
- [3] Christian Grimm, Marc Neitzner, Helmut Pralle, Jens-S. Vockler
"Request Routing in Cache Meshes", 3rd international WWW Caching Workshop, June 1998
- [4] A. Chankuntod, P. B. Danzig, C. Neerdaels
"A hierarchical internet object cache", Proceedings of 1996 USENIX, 1996.
- [5] Vinod Valloppillil, Keith W. Ross
"Cache Array Routing Protocol",
<http://egg.microsoft.com/carp/carpwp.doc>
- [6] 上谷 一, 太田 耕平, 加藤 寧, グレン マンスフィールド, 根元 義章
"構成情報を利用したネットワーク利用の効率化", 1998年3月 電子情報通信学会総合大会
- [7] Hajime Kamiya, Kohei Ohata, Nei Kato, Glenn Mansfield, Yoshiaki Nemoto
"Improving efficiency of network services", APANOMS'98, September 1998