IPv6におけるサイトローカルアドレスのステートレス自動設定

島田 将行[†], 吉廣 卓哉[‡], 岡部 寿男[‡], 岩間 一雄[‡]

IPv6のサイトローカルアドレスを無設定(zeroconf)でステートレスに自動割り当てする枠組を提案する。当該セグメントに最初に接続されたルータが、そのセグメントのリーダとなって、16bitのsubnet ID部分をランダムに選んだプレフィックスを生成する。アドレスのサイト内での一意性を保証するために、RIPngを拡張したプレフィックスアドレス重複の検出方法を示す。重複が検出されると直ちにセグメントリーダへ通知され、IPv6のアドレス付替の仕組を用いて新たなアドレスへの付替が行われる。

Stateless auto-configuration of IPv6 site-local addresses

Masayuki Shimada[†], Takuya Yoshihiro[‡], Yasuo Okabe[‡], Kazuo Iwama[‡]

We propose a new framework for stateless auto-configuration of IPv6 site-local addresses with "zero configuration." When a router is connected to a segment as the first router, it becomes the segment leader and generates an address prefix with randomly generated 16-bit subnet ID for the segment. In order to assure the uniqueness of the prefix addresses, we show a protocol, which is an extension of RIPng, for detecting prefixes conflicts. When a conflict is detected, it is notified to its segment leader immediately. The segment leader generates a new prefix and renumber to new one via IPv6 address renumbering scheme.

1 はじめに

近年のインターネットの普及やデジタル家電の登場により、TCP/IPを用いた家庭内ネットワーク実現への期待が高まっている。このようなネットワークでは管理者による設定やメンテナンスを望めないため、アドレス取得時に現状のDHCPを用いることができず、それらを完全に自動化するプロセスが必要となる。

単一リンク内のアドレス自動設定としては、これまでにIPv4におけるAuto-IP[12]やZeroconf IPv4 link-local address[5]、IPv6におけるリンクローカルアドレスのステートレスアドレス自動設定[10, 11]が提案されているが、複数のネットワークセグメントにインターフェースを持つルータまで含めたマルチセグメントの場合は対象外である。IPv6におけるステートレスアドレス自動設定[10, 11]では、端末ホストがルータからの情報を用いてルータを跨いで使用可能なアドレスプレフィックスを自動的に決定できるが、ルータには予めこれらのプレフィックスを設定しておかなければならない。

マルチセグメントを対象とする無設定(zeroconf)のアドレス自動設定については、IETFのZEROCONF WGで最近になって検討がはじまった[3]。これまでに、mini-DHCP[1, 2]およびZRIP[4]により、IPv4におけるプライベートアドレスのランダム割当とそれらの競合検出、その解決プロセスが提案されている。mini-DHCPでは、ルータによるランダムなアドレス決定とホストへの通知方法が規定されている。またZRIPでは、各セグメントを一意に識別するIDとアドレスを１つの組として扱い、RIPルーティングデーモンの取り交わすメッセージを用いた競合検出とその解決プロセスが規定されている。

本研究では、IPv6におけるサイトローカルアドレス[8]のランダム割当とそれらの競合検出、その解決プロセスを提案する。IPv6では、ステートレスアドレス自動設定の枠組を用いてそれを拡張することにより、mini-DHCPのような新しいプロトコルを用いずにアドレスの決定とホストへの通知が行えるという利点がある。またIPv6のアドレス有効期限の機能を用いることで、アドレス変更時もそれまでに古いアド

[†]京都大学工学部情報学科
Undergraduate School of Informatics, Kyoto University
[‡]京都大学大学院情報学研究科
Graduate School of Informatics, Kyoto University

レスを用いて張られているコネクションを保つことができる。

さらにIPv6における競合検出のプロトコルとして、RIPngにZRIPと同等の拡張を加えたZRIPngを提案する。競合検出は、経路情報としてプレフィックスアドレスとともにセグメントリーダのUID (Unique Interface Identifier) を伝播させることで行う。ZRIPのZRIPngとの大きな違いは、競合検出の報知を経路制御プロトコルとは切り離したことである。これによりRIPng以外のプロトコルに対しても同様の拡張を考えることができる。

また、提案する枠組の有効性を検証するための実装を、現在作成中である。ZRIPngのルーティングデーモンに必要なUIDを用いた競合の検出とアドレス変更後のルート情報の伝播制限は、GNU Zebra[13]のripngdを拡張することで実装している。

以下、2章でIPv6に関する基本的な概念について述べ、3章で対象とする無設定ネットワークについて述べる。4章で本研究で提案するサイトローカルアドレスのステートレス自動設定の枠組とそのためのIPv6の拡張について示し、5章で競合検出のプロトコルであるZRIPngの提案を行う。6章では実装へ向けての検討事項について示す。

2 基本概念

2.1 IPv6サイトローカルアドレス [8]

IPv6における集約可能なグローバルユニキャストアドレスおよびローカルなユニキャストアドレスは、上位64bitのプレフィックスと下位64ビットのインターフェースIDからなる。インターフェースIDは、IEEE 48bit MACアドレスなどから導かれる当該リンクで一意なビット列である。プレフィックスはアドレスの種類を区別するformat prefixと呼ばれるビット列から始まる。本研究で対象とするサイトローカルアドレスの場合format prefixの値がFEC0/10であり、これに続く38bitの0と16bitのsubnet IDによりプレフィックスが構成される。

2.2 ステートレスアドレス自動設定 [11]

IPv6では、起動したホストはまず、リンクローカルアドレスプレフィックス (FE80::/64) とインターフェイスIDからリンクローカルアドレスを生成し、このアドレスを対象アドレスとして、近隣請求メッセージ (Neigbor Solicitation Message) を送信することで、個別性を確認する。

リンクローカルアドレス決定後は、定期的な送信を待つか、あるいはすぐに送信させるためにルータ請求メッセージ (Router Solicitaion Message) を送信してルータ通知メッセージ (Router Advertisement Message) を受け取り、それに含まれる情報を用いてアドレスやその他の通信パラメータの設定を行なう。ルータ通知メッセージを受信する事ができなかった場合 (セグメント内にルータが存在しなかった場合)、ステートフル自動設定を用いなければならない事になっている。

2.3 アドレスの有効期限 [10]

IPv6では、どのアドレスについても有効期間が定められている。有効期間が満了すると、アドレスとインターフェイスの結び付きが無効となり、アドレスを別のインターフェイスに割り当て直すことができる。このような結び付きをサポートするために、割り当てられるアドレスには、アドレスを自由に利用できることを示す「推奨 (preferred)」段階と、もうすぐアドレスが使えなくなることが予想されるために、これ以上の使用は認められないことを意味する「有効期限切れ (deprecated)」段階との2段階が定められている。「有効期限切れ」段階に設定されたアドレスは、そのアドレスを用いて継続中の通信に関しては使用できるが、新たなコネクションを張るときには使用せず、徐々に使用を止めなければならない。

3 無設定ネットワーク

本研究で想定するのは、TCP/IPネットワークの特殊なクラスでありIETF Zeroconf WGが対象としている無設定 (Zeroconf) ネットワーク [7] である。

無設定ネットワークでは、家庭内ネットワーク等、管理者の存在しないネットワークがまさにそうであるように、ユーザによる設定や集中管理されたサーバからの情報を期待することはできない。そこではアドレスの取得にDHCPサーバを用いたり、ルータ広告で伝えられるプレフィックスが人為的に設定されたりすることはない。

本研究では、複数のネットワークセグメントがルータによって接続されたmulti-router zeroconf network [3]を考える。各セグメントはブロードキャストネットワークであり、かつルータのインターフェースIDについてはサイト内での一意性が保証されているもの(以下「Unique interface identifier (UID)」と呼ぶ)とする。

対象とするネットワークでは、通常状態でのノード(ホストまたはルータ)の参加および離脱だけでなく、ノードの故障と復旧、リンクの切断と接続などが起こり得る。

4 サイトローカルアドレスの自動設定

4.1 ノードの起動

本研究で提案するサイトローカルアドレスのステートレス自動設定では、ネットワークに参加するノードはまず、通常のステートレスアドレス自動設定によってリンクローカルアドレスを決定する。さらにそのノードがルータであっても、ルータ通知メッセージを受け取った場合にはそれに従うものとする。

ルータ通知メッセージによってサイトローカルアドレスを決定することができない場合、本プロトコルで規定されるサイトローカルアドレス自動設定が有効化されているルータであれば「セグメントリーダ」になる。セグメントリーダは、当該セグメントに対するサイトローカルアドレスプレフィックスの管理や他ノードへの伝達に関し責任を負う特別なルータであり、セグメント内に原則として高々1つ存在する。セグメントリーダは当該セグメントに割り当てるサイトローカルアドレスプレフィックスを(16bitのsubnet IDをランダムに)決定する*。以後に新たなノードが現れた時には、ルータ通知メッセージによってこのプレフィックスを通知する。

加えて、当該セグメントのアドレスに対し他のルータが競合を検出した時に備え、競合検出メッセージを待ち受ける。

4.2 アドレスの変更

アドレスの競合などにより、アドレスの付け替えが行なわれるときでも、古いアドレスを用いてすでに確立されているコネクションが突然切れてしまうことは好ましくない。セグメントリーダは新しいアドレスの使用開始後、古いアドレスを有効期限切れ段階に設定し、セグメント内の他ノードにこのアドレスを通知する際にもそれを反映させる。古いアドレスは一定期間後に無効化される。

4.3 セグメントリーダの生存確認

セグメントリーダはアドレスプレフィックスの管理上重要な役割を担っており、それらが何らかの障害により機能しなくなった際にはすぐに新しいセグメントリーダを選出し直す必要がある。セグメント内のノードは、セグメントリーダが定期的に送出するルータ通知メッセージを受け取ることでセグメントリーダが動作していることを確認する。定期的なルータ通知メッセージの受信が途絶えた時、当該セグメントの中に他にルータが存在すれば、これらのルータの中から新たなセグメントリーダが選ばれる。新しいセグメントリーダの決定とともに、それまで用いられていたアドレスは変更される。

一方、何らかの事情で単一のリンクに複数のセグメントリーダが存在する状態になったときは、定期的に送出されるルータ通知メッセージによりそれら全てのセグメントリーダがそのことを知る。所定の手順でそのうち1つが選出され、残りはセグメントリーダでなくなりアドレスを解放する。

*この際、すでにセグメントリーダの経路表上にあって競合することが明らかなsubnet IDは除外する。

5 競合の解決

5.1 競合の検出

ランダムに決定されたサイトローカルアドレスプレフィックスは、ネットワーク間の新たな接続などにより、競合を生じさせる可能性がある。本研究で提案するZRIPngでは、ZRIP[4]と同様に、ランダムに決定されたアドレスプレフィックスとセグメントリーダのUIDとを対にして経路情報として伝播させ、同じプレフィックスに対して異なるUIDを持つ組があるかを調べることでアドレスの競合を検出する。

5.1.1 UIDの伝播

UIDを経路情報のエントリに含めて伝播させるためには、経路制御プロトコルを独自に拡張する必要がある。しかし既存のルーティングプロトコルやその実装をでき得る限りそのまま利用できるようにするために、メッセージフォーマットの変更は避けることが望ましい。そこで、RIPngやOSPFv3のようなIPv6用の経路制御プロトコルのプレフィックス情報フィールドが128bitである点に注目し、このフィールドにUIDを含めて伝播させることを考える。サイトローカルアドレスを用いたセグメント間のルーティングでは、そのプレフィックス部分である上位64bitの情報で十分であり、下位64bitは必ず0で埋められている。われわれはこのビット列にUIDを含めて伝播させることを提案する。今回は特にRIPngに注目し、それを拡張することによる実現方法について述べる(図1)。ここで提案するプロトコルを**ZRIPng**と呼ぶ。

5.1.2 UIDの比較

ルータは、経路エントリの中に、受け取ったルート情報エントリとデスティネーションアドレスが一致し、UIDが異なるエントリがないかを調べる。もしそのようなエントリが存在すれば、エントリのデスティネーションとなっている二つのセグメントの間で競合が生じていることになる(図2)。

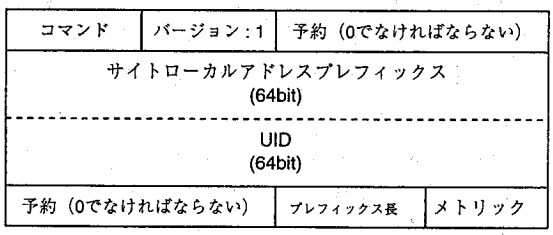

図 1: RIPngのRTEに含まれたUID

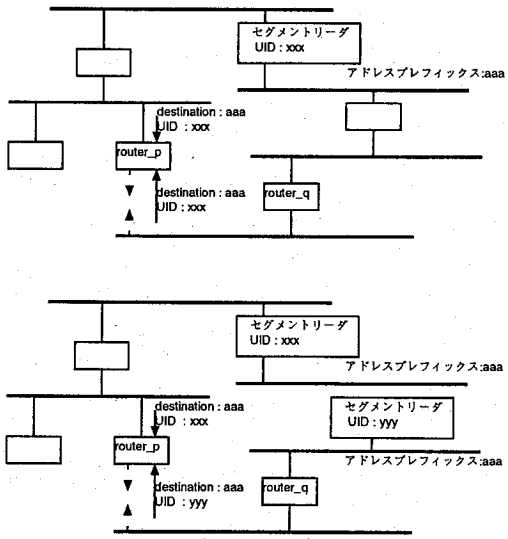

図 2: 競合検出

5.2 競合の解決

競合検出のプロセスに比して、その解決プロセスにおいてはZRIPと異なる方法を採用している。すでに述べたように、一つはアドレス付け替えを要求するために明示的なメッセージの送信を行なう点、もう一つは既存のコネクションに対して一定期間接続を保とうとしている点である。

5.2.1 競合検出メッセージの送信

競合を検出したルータは、競合対象となるアドレスを使用しているセグメントのうち、先に知っていた方のセグメントリーダに対して競合が検出されたことを報せるメッセージを送信する。以後、このメッセージを競合検出メッセージと呼ぶ。競合検出メッセージの詳細は決定していないが、Router Renumbering[6]のようなICMPv6の拡張として新たに規定するのがふさわしいと考えている†。

5.2.2 ルート情報の伝播制限

新しいアドレスと古いアドレスを矛盾の無いものとして扱うために、競合を検出したルータ側ではルート情報の伝播について以下のような制限を加える。

- 競合検出の対象とならなかったアドレスに対する経路情報は通常通り伝播させる
- 競合検出されたアドレスに対する経路情報は、競合する情報を送ってきたインターフェース方向へは伝播させない
- 競合検出のきっかけとなるルート情報を受け取っていないインターフェイスに対しては、競合対象となるルート情報のうち、当該ルータが最初に知っていた(先に受け取った)方のルート情報のみを伝播する。

3番目のルールは、検出したルータが3つ以上のインターフェイスを持つマルチルータであったときのためのルールである。ここでは競合を検出したルータが3つのインターフェイスを持つマルチルータであったとして、具体的にどのようにルート情報を伝播させるかについて説明する。

図3は、ルートテーブルエントリのネクストホップがiface_aであるルート情報route_aを知っていたルータが、インターフェイスをiface_bから競合するルート情報route_bを受け取った時の図であり、残りのインターフェイスをiface_cとしている。

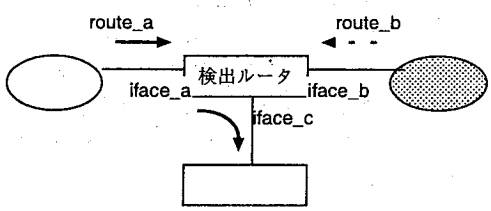

図3: ルート情報のフィルタリング

まず、route_aはiface_bに対しては伝播させない。これにより1つの競合を複数のルータが次々に検出してしまうことを避ける。同じ理由からroute_bはiface_aには流さない。また、水平分割[9]により、route_aはiface_aには流されず、route_bはiface_bには流されない。残るはiface_cに流す情報であるが、このインターフェイスに対してはroute_aのみを流す。競合が検出されるまでにiface_cの先にあるノードに伝播させていたのはroute_aのみであるはずであり、それらのノードが確立しているコネクションを維持するためにはroute_aのみで十分である。

6 実装

提案する枠組の有効性を検証するための実装を、現在作成中である。

まず、OSレベルで実装されているIPv6プロトコルスタックにおいてステートレスアドレス自動設定に関わる部分の変更が必要である。これは、FreeBSD上のIPv6実装であるKAME[14]を改造することによって実装している。

また、セグメントリーダとしての機能であるサイト

†現状は独自のUDPメッセージとして送信することで実装を行なっている。

ローカルアドレスをランダムに決定や各インターフェイスへの割当、有効期限設定、競合検出メッセージの待ち受けと、それに付随するセグメントリーダの決定や生存確認機構などは、そのためにまとめて1つのデーモンとして実装しようとしている。

ZRIPngでの経路制御に必要なUIDを用いた競合の検出とアドレス変更後のルート情報の伝播制限は、GNU Zebra[13]のripngdを拡張することで実装している。ZebraのZRIPng拡張部分では、検出済の競合について、競合アドレスやUID、経路エントリの受信インターフェイスなどの情報を保持するためのテーブルをもたせる。経路エントリの送受信時にはそれを照合することで、競合の検出とルート情報のフィルタリングを行う。

7 おわりに

無設定(zeroconf)の環境で、IPv6のサイトローカルアドレスをステートレスに自動割り当てし、その一意性保証を行う枠組を提案した。実装の完成を急ぎ、実環境で有効性と限界を検証することが課題である。

現在の仕様では、競合が生じたアドレスが接続後もしばらく使用できない点に、改善の余地が残っている。また、アドレスの集約や複数のルーティングプロトコルの併用などについても、今後検討していきたい。

参考文献

[1] Bernard Aboba "Auto-Addressing in Multi-segment Networks", draft-aboba-zeroconf-multi-00.txt, INTERNET-DRAFT, Zero-conf WG, October 6 1999.

[2] Akinlar & Braun & Mukherjee, "Mini-DHCP Election Option for DHCP", draft-akinlar-zeroconf-minidhcp-option-00.txt, INTERNET-DRAFT", Zero-conf WG, March 5 2000.

[3] Cuneyt Akinlar & David Braun & Sarit Mukherjee, "Multi-Router Zeroconf Network Requirements", draft-akinlar-zeroconf-multirouter-01.txt, INTERNET-DRAFT, NETWORK WG, August 15 2000.

[4] Akinlar & Braun & Mukherjee, "Zero-Conf Routing Information Protocol", draft-akinlar-zeroconf-zrip-00.txt INTERNET-DRAFT, Zero-conf WG, August 2000.

[5] Stuart Cheshire, "Dynamic Configuration of IPv4 link-local addresses", draft-ietf-zeroconf-ipv4-linklocal-01.txt, INTERNET-DRAFT, Zero-conf WG, 30th November 2000

[6] M. Crawford & Fermilab, "Router Renumbering for IPv6", RFC 2894, Standards Track, August 2000.

[7] M.Hatting, "Zeroconf Requirement", draft-ietf-zeroconf-reqts-06.txt, INTERNET-DRAFT, Zero-conf WG, November 20 2000.

[8] R.Hinden & S.Deering, "IP Version 6 Addressing Architecture", RFC 2373, July 1998.

[9] G.Malkin & R.Minnear, "RIPng for IPv6", RFC 2080, January 1997.

[10] T.Narten & E.Nordmark & W.Simpson, "Neighbor Disovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[11] S.Thomson & T.Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.

[12] "Universal Plug and Play Device Architecture", Microsoft Corporation,8 Jun 2000.

[13] GNU Zebra —routing software, URL http://www.zebra.org/docs.html

[14] KAME Project, URL http://www.kame.net/