マルチメディア通信と分散処理 71−23
（1995．7．14）

次世代インターネットプロトコル(IPv6)に関する一考察

南 政樹 * 冨永明宏 † 寺岡文男 ‡ 村井 純 §
慶應義塾大学 慶應義塾大学 ソニー CSL 慶應義塾大学

1995年7月13〜14日

近年のインターネットに対する関心の高さを象徴するように，インターネットは急激に成長し新たに社会的な要求を受けるようになってきた．そのため，既存の技術では対応できない部分も多くなり，それを考慮した新しいインターネットの基盤が必要とされている．インターネットの最も基礎的な技術であるインターネットプロトコル(IP)は，今まさに次世代へ向けて進化しようとしている．

本稿では次世代インターネットプロトコルを現在のインターネットプロトコルと比較・考察し，どのようにそれらの問題が解決されようとしているのかを述べた後に，実装について考察を述べる．

Internet Protocol for Next Generation(IPv6)

Masaki Minami Akihiro Tominaga Fumio Teraoka Jun Murai
Keio University Keio University Sony CSL Keio University

Since people's interest about Internet is rising in these days, Internet has grown up abruptly, and is newly required of social demand. Therefore, existing technology found it difficult to cope with that rapid growth ,then taking into consideration about everything, new Internet inflastructure is needed . The most basic technology in the field of Internet, IP is about to change for the next generation.

In this thesis, it is considered how IP for the next generation(IPng:IPv6) can replace present IP (IPv4),and how they cope with both technical and social problems caused by replacing. And then, we describe how the implementation is going.

1 背景

近年，インターネットは次世代の情報社会基盤として社会的に注目されている．事実，ここ数年の間でインターネットは急激な成長を遂げている．インターネットに接続されているホスト数は指数関数的な増加傾向を示しており，今後は更にその増加率が大きくなることが予想されている．

ところが，このインターネットを支える基礎的な技術基盤である，インターネットプロトコル(IP)は，約20年前に開発されたほぼそのままの形で現在も運用されており，開発当時の予想を上回るこの規模の変化は，様々な問題を引き起こしている．例えば，IPアドレス空間の枯渇問題や経路制御情報の氾濫などである．

IPがこの20年間ほとんど変化しなかったのとは対照的に，インターネットを取り巻く計算機や通信媒体などの環境は大きく変化してきた．例えば，計算機の小型化，高性能化，廉価化，また通信媒体の高品質化，大容量化，高速化などである．これらの事によって，人間は計算機の利用形態を変化させ，インターネットは新たな要求を受けるようになる．それは，例えば人間は計算機を持ち歩き，移動した先でネットワークと接続するといった移動端末のサポートであったり，VoD(Video on Demand)などで動画や音声の通信を行うために実時間性のサポートである．

こうした問題や要求は現在のインターネットプロトコルでは，根本的な解決ができない．インターネット全体の技術的な問題を解決するための組織であるIETF(Internet Engineering Task Force)では，この問

*minami@sfc.wide.ad.jp
†tomy@sfc.wide.ad.jp
‡tera@csl.sony.co.jp
§jun@sfc.wide.ad.jp

題を解決するために新たなインターネットプロトコルを提案，議論を行ってきた．いくつかの候補の中で現在の IP を簡素化しアドレス空間を 128bit に拡張した SIPP(Simple Internet Protocol Plus) が次世代 IP (IP for Next Generation:IPng) として決定した．

この次世代 IP は IANA(Internet Address Numbers Authority) から IP Version 6 として，新しい IP のバージョン番号が与えられている．本文中ではこれを IPv6 と呼ぶことにする[1]．これに対して現在の IP は Version 番号が 4 である．これも同様に IPv4 と呼ぶことにする．

2 IPv6 の概要

IPv6 は IPv4 から発展した形で考案されたおり，基本的な性質，機能は同じである．ここでは具体的に IPv6 について現在の IPv4 と比較して述べる．

2.1 ヘッダフォーマット

IPv6 は IPv4 のヘッダフィールド (図 1) のいくつかを廃止，もしくは拡張 (オプション) ヘッダ (2.2章) を付加し，オプションとして指定することにより，標準のヘッダフォーマットを簡素化し，パケット処理のコストを低減するように設計されている．簡素化された IPv6 のフォーマットは図 2のような構成になっている．

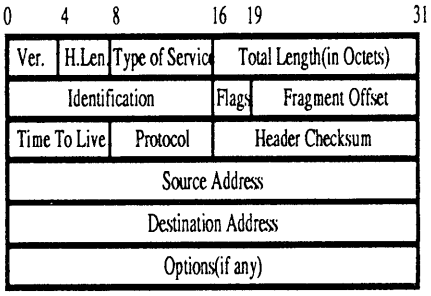

図 1: IPv4 ヘッダフォーマット

IPv4 ではヘッダが可変長のため "H.Len"(Header Length) でヘッダの大きさを定義している．IPv6 では IPv6 ヘッダ自身は固定長のため，このフィールドは必要ない．また同様の理由から，IPv6 では，"Total Length" などでパケット全体の大きさを示すではなく，データ部の大きさを示す "Payload Length" に変更されている．

[1]現在，世界中で様々なアーキテクチャに対し IPv6 の実装が行われている．もうすでにいくつかの実装例がある．我々も BSD 系の UNIX で実装を始めている．

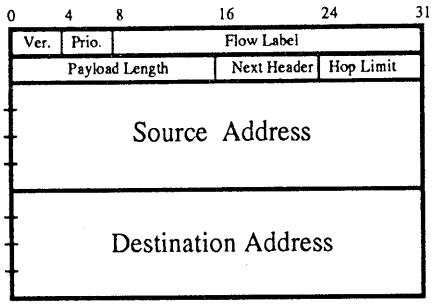

図 2: IPv6 ヘッダフォーマット

"Ver."(Version) フィールドは，インターネットプロトコルのバージョンを示す 4bit のフィールドで IPv4 では "4" が，IPv6 では "6" が設定される．

Number	Categories
0	uncharacterized traffic
1	"filler" traffic. (e.g. NetNews)
2	unattended data transfer (e.g. e-mail)
3	(reserved)
4	attended bulk transfer (e.g. FTP, NFS)
5	(reserved)
6	interactive traffic (e.g. telnet, X)
7	internet control traffic (e.g. routing protocols, SMTP)

表 1: 輻輳制御下での優先順位

"Prio."(Priority) はそのパケットが配送される時に期待されている優先度が 4bit で表現される．この値は，そのパケット転送が輻輳制御下にあるかどうかで 0-15 のうち，0-7，8-15 で分かれる．輻輳制御が有効なデータの転送の場合 0-7(表 1) の数値が，輻輳制御が有効でないデータ転送においては 8-15 の数値が用いられる．後者の場合，数値の小さい方が輻輳状態でパケットを破棄しても良い度合が高い．

"Flow Label" フィールドは，QoS の制御やリアルタイムサービスのためのフィールドである．このフィールドは 24bit である．IPv6 ではこの "Flow Label" フィールドによって，タイプに応じた転送方法を定義できる．ちょう

どこれと対照的なのがIPv4の“Type of Service”(ToS)フィールドでである．ところが，現在のインターネットにおいて，このToSを用いることは滅多になく，実装すらされていないことが多い．

“Next Header”フィールドは，IPv6ヘッダに続くオプションヘッダ，もしくは上位層のプロトコルのタイプを指定するために利用する8bitのフィールドである．上位層のプロトコルタイプのを示す値は，IPv4のプロトコルフィールドと同じ値を使用する．

“Hop Limit”は，IPv4のTTLフィールドの機能と類似しているが，正確にはTTLはパケットを転送するホストに保持されていた秒数分減らさなければならない．これに対してIPv6の“Hop Limit”は単純なホップ数を表している．

“Source Address”は，データを送る側，始点アドレスを指定するフィールドで，“Destination Address”はデータを受け取る側，終点アドレスを指定する．

2.2 拡張ヘッダ(オプションヘッダ)

IPv6では，ネットワーク層で必要な付加的な情報を，拡張ヘッダとして，IPv6ヘッダとは切り離した形で，パケットに含めることができる．IPv4では，ネットワーク層で必要な付加的な情報はIPヘッダの内部に持っていた．これは非常に複雑で，パケットを処理する度にIPヘッダとデータ部を分けていかなくてはいけないので非常に非効率的にもみえる．

この場合IPv6の拡張ヘッダは非常に柔軟なオプションの設定を可能にしている．また，新たな機能を加える時も非常に簡単に対応でき，拡張性にも優れている．

現在考案されている，拡張ヘッダの種類は次の通りである．

- Hop-by-Hopオプションヘッダ
- Routingヘッダ
- Fragmentヘッダ
- Destinationオプションヘッダ
- Authenticationヘッダ
- Encapsulating Security Payloadヘッダ

2.2.1 Hop-by-Hop optionヘッダ

Hop-by-Hopヘッダは，パケットが通過するすべてのノードで処理されるヘッダであり，可変長のフォーマットを持ち複数のオプションを指定できる．オプションフィールドはTVL(Text Length Variable)というフォーマットで記述される．このオプションはIPv6ヘッダのPayloadよりも大きなデータを転送する時に“Jumbo Payload Length”が含まれて使用される．

2.2.2 Routingヘッダ

routingヘッダは，IPv4のSource Routeオプションと類似した機能を持っている．このヘッダはパケットの通過経路のノードが付加すべきものではなく，送信元がルーティングに必要な情報をすべて付加する必要がある．ルーティングのタイプは複数指定できるようになっているが，現在規定されているのは，ルーズソースルーティングのみである．

2.2.3 Fragmentヘッダ

Fragmentヘッダは，パケットの送信元が受信先の経路のMTUよりも大きなデータを送ろうとする場合に指定するヘッダである．同じパケットから分割されたパケットは，共通のFragment IDを持っており，受信先ではこれを利用して分割されたデータを再構成する．IPv6では，IPv4とは異なり，中間ノードでのフラグメントは行なわれない．これは，IPv4ではそれほど使用されてなかったPath MTU discoveryという技術を用いることにより，MTUの検出を行なうことができるからである．

2.2.4 Authenticationヘッダ

Authenticationヘッダは，認証情報をパケットに付加することでセキュリティを提供する機能を持つ．Authenticationヘッダにより IP レベルでのセキュリティ機能が強化されたが，パケットモニタリングなどによるトラフィック解析に対するセキュリティは考慮されていない．この問題はかなり議論されており，結論としては「やるべきである」ということになっているが，現在のところ実装されている例はない．

2.2.5 Destination optionヘッダ

Hop-by-hopヘッダと同様のフォーマットを持つが，Destinationヘッダで指定されたオプションはパケットの受信先でのみ有効となり，中間のノードでは無視される．

2.3 オプションヘッダのフォーマット

通常，IPv6のパケットもIPv4のパケットと同じようなパケットのフォーマット(図3参照)でデータ転送が行われるが，拡張ヘッダが付けられた時は，IPv6ヘッダと上位層のヘッダの間，つまり，IPv6ヘッダとデータ部の間にオプションヘッダを格納して(図4参照)データ転送を行う．オプションヘッダが複数付加された場合(図5参照)も同様である．

複数のヘッダを指定する場合，ヘッダが現れる順番は次のような順番でなければならない．

1. IPv6 header

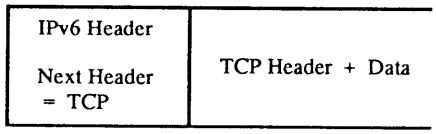

図 3: 拡張ヘッダのない IPv6 パケット

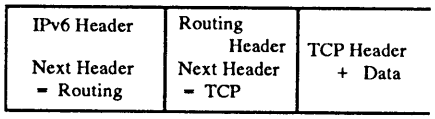

図 4: IPv6 拡張ヘッダの例 (1)

2. Hop-by-Hop Options header
3. Destination Options header
4. Routing header
5. Fragment header
6. Authentication header
7. Encapsulating Security header
8. Destination Options header
9. upper layer header

2.4 アドレスアーキテクチャ

IPv6 は，IPv4 の大きな課題であったスケーラビリティの問題を解決するために，128 ビットのアドレス空間を設定した．IPv4 では 32bit のアドレス空間で，約 43 億のホストとネットワークの識別が可能であったが，128bit になり約 3.4×10^{38} のアドレスを提供している．世界中の人口一人あたり約 10^{28} 個のアドレスを，地表面 $1cm^2$ あたり，10^{20} 個のアドレスが利用可能である．

IPv6 のアドレスには，ユニキャストアドレス (Unicast Address)，エニイキャストアドレス (Anycast Address)，マルチキャストアドレスの 3 種類が存在する．ブロードキャストアドレスは，アドレスの種類としては定義されず，特別なマルチキャストアドレスと定義される．

また，IPv4 のクラスと同じように，アドレスの上位数 bit でアドレスのタイプを分けて表現できるように定義されている．つまり，アドレスの始めの数 bit を解析することで，そのアドレスがどのタイプのアドレスなのか調べることができる．表 2に，そのアドレスのタイプと意味をまとめる．

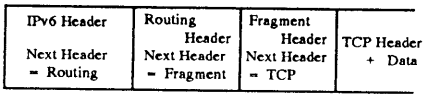

図 5: IPv6 拡張ヘッダの例 (2)

識別子	タイプ
0000 0000	Reserved
0000 0001	Unassigned
0000 001	Reserved for NSAP Allocation
0000 010	Reserved for IPX Allocation
0000 011	Unassigned
0000 1	Unassigned
0001	Unassigned
001	Unassigned
010	Provider-Based Unicast Address
011	Unassigned
100	Reserved for Neutral-Interconnect-Based Unicast Addresses
101	Unassigned
110	Unassigned
1110	Unassigned
1111 0	Unassigned
1111 10	Unassigned
1111 110	Unassigned
1111 1110 0	Unassigned
1111 1110 10	Link Local Use Addresses
1111 1110 11	Site Local Use Addresses
1111 1111	Multicast Addresses

表 2: アドレスタイプ

Provider-based Address は，インターネットプロバイダが提供するアドレス空間を示している．NSAP，IPX 用のアドレスは，IPv6 のアドレスと NSAP, IPX のアドレス体系との 1 対 1 のマッピングを実現するためのものである．

Local Use Address は，インターネットに接続しないサイトが内部のノードの識別に使用するアドレスである．

2.5 ユニキャストアドレス

CIDRを利用しているIPv4の上でのユニキャストアドレスと構造は似ている．図6ではその一例として，インターフェイスアドレスにEthernetのMACアドレスを使う例を考える．

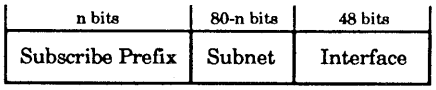

図 6: IPv6のユニキャストアドレスの構造

最初の “n” bitの “Subscriber Prefix” と “80-n” の “Subnet ID” はそのリンクに接続されているルータから得られる情報である．このような仕組みは，移動ホストに対するIPアドレスの自動設定機構などで有効であると考えられている．

2.6 エニイキャストアドレス

エニイキャストアドレスは，ユニキャストアドレスと同様に，1対1の通信に使用するアドレスであるが，特定のノードを指定するものではない点が異なっている．エニイキャストアドレスは，同一識別子を持つアドレス群の中の一つのノードを指定ずる．エニイキャストアドレスは，例えばポリシールーティングなどでの使用が想定されている．

2.7 マルチキャストアドレス

IPv6のマルチキャストはIPv4のそれを拡張した形で提供される．マルチキャストアドレスは，図7ようなフォーマットが定義されている．

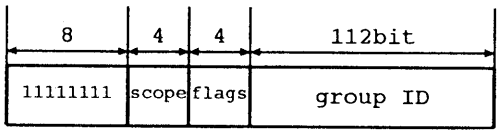

図 7: マルチキャストアドレスフォーマット

上位の8ビットは先に述べた識別子である．flagsは4ビットのフィールドで上位3ビットは現在はまだ未定義で常に0にセットされる．最下位ビットが0の場合は，そのマルチキャストアドレスが恒久的に割り当てられたものであること(インターネット全体で ”well-known”であること)を示している．最下位ビットが1の場合は，マルチキャストアドレスが一時的なものであることを示している．

scopeフィールドは4ビットのフィールドでマルチキャストの範囲を指定する．現在規定されているscopeフィールドの値は3のようなものである．

0	reserved
1	intra-node scope
2	intra-link scope
3	(unassigned)
4	(unassigned)
5	intra-site scope
6	(unassigned)
7	(unassigned)
8	intra-organization scope
9	(unassigned)
A	(unassigned)
B	(unassigned)
C	(unassigned)
D	(unassigned)
E	global scope
F	reserved

表 3: scope フィールド

3 まとめ–実装へ向けて–

現在，我々のグループではBSD系のUNIXでIPv6の実装を行っている．この実装で必要とされている要求はいくつかあるが，我々はまずIPv4のパケットとIPv6のパケットの両方を，同じインターフェイスで処理できるようにすることが最も重要であると考える．また，拡張ヘッダの取り扱いもIPv4の実装には含まれていないので，新たに実装する必要がある．

そこで我々は，できるだけ今のBSDのコードを残す形で，さらに利用できるIPv4は最大限利用しつつIPv6を実装できないか考察してみた．

3.1 IPv4とIPv6の共存

そのパケットがはたしてIPv4かIPv6かをどこで見分ければよいのか，まずそれが問題である．EthernetFrameの中のネットワーク層のプロトコルを指す2byteの値はIPv4もIPv6も同じ “0x0800” と規定されているためこ

れを利用してどちらの解析モジュールを呼び出すかと言う判断は不可能である.

そこで, I/F を初期化する際に「IPv4 で初期化するか」「IPv6で初期化するか」を決め, もし両方のパケットを受ける必要があれば, 一旦初期化した I/F に対し別名でもう一度定義する方法を考えた. すなわち, 一つの I/F に二つのアドレスをつけることになる.

実際に, IPv4 のアドレスと, XNS プロトコルのアドレスを同一のインターフェイスに与えることは可能である.

3.2 拡張 (オプション) パケットの解析

IPv4 では拡張 (オプション) ヘッダの解析はやっていないものの, 上位層へデータ部を渡す際に, 上位層にあるプロトコルが何かを判断して, 呼び出す関数を変えている. IPv4 では protosw 構造体配列 (図 8) に, それぞれの関数へのポインタを保持し, 配列の位置と構造体のメンバを指定してそれぞれの関数にデータ部を渡している.

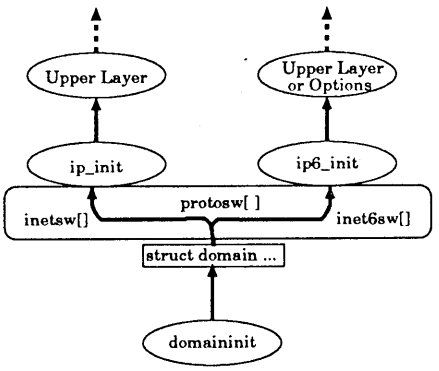

図 8: protosw の構造

我々はこれを利用し, 新たにドメインとアドレスファミリ, プロトコルファミリ (図9) を定義することで IPv4 とは完全に別の IPv6 のモジュール群を作っていく方法を考えた. モジュールへデータを渡す場合に必要となるので拡張ヘッダは一旦 mbuf に IPv6 ヘッダとは分割して保持することにする. この方法では UNIX のカーネルが大きくなることや, mbuf に関する操作が多くなることが予想されるが, いまは IPv4 と IPv6 との共存を考えることを優先する.

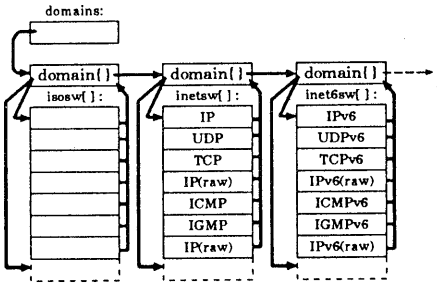

図 9: domain の構造

3.3 今後の予定

今後は実装を行い, 現在公開されている NetBSD の実装との相互接続性のテストを行っていく予定である.

4 謝辞

本論文を書くにあたり, 慶應義塾大学環境情報学部助手 中村 修氏, 東京大学大型計算機センター 加藤 朗氏, 慶應義塾大学環境情報学部付属環境情報研究所 植原啓介氏, 慶應義塾大学政策・メディア研究科 2 年の西田佳史氏, の諸氏は, 本論文の初期の段階からお世話になった. また, WIDE プロジェクト NewArc-WG の方々からも様々な助言を頂いた. この場をお借りし感謝の意を表します.

5 参考文献

参考文献

[1] W. Richard Stevens *TCP/IP Illustrated, Volume 1, The Protocols* Addison-Wesley, 1994

[2] Gary R. Wright, W. Richard Stevens *TCP/IP Illustrated, Volume 2, The Implementation* Addison-Wesley, 1995

[3] S. Deering, R. Hinden *Internet Protocol, Version 6 (IPv6) Specification* draft-ietf-ipngwg-ipv6-spec-02.txt

[4] R. Hinden, S. Deering *IP Version 6 Addressing Architecture* draft-ietf-ipngwg-addr-arch-03.txt