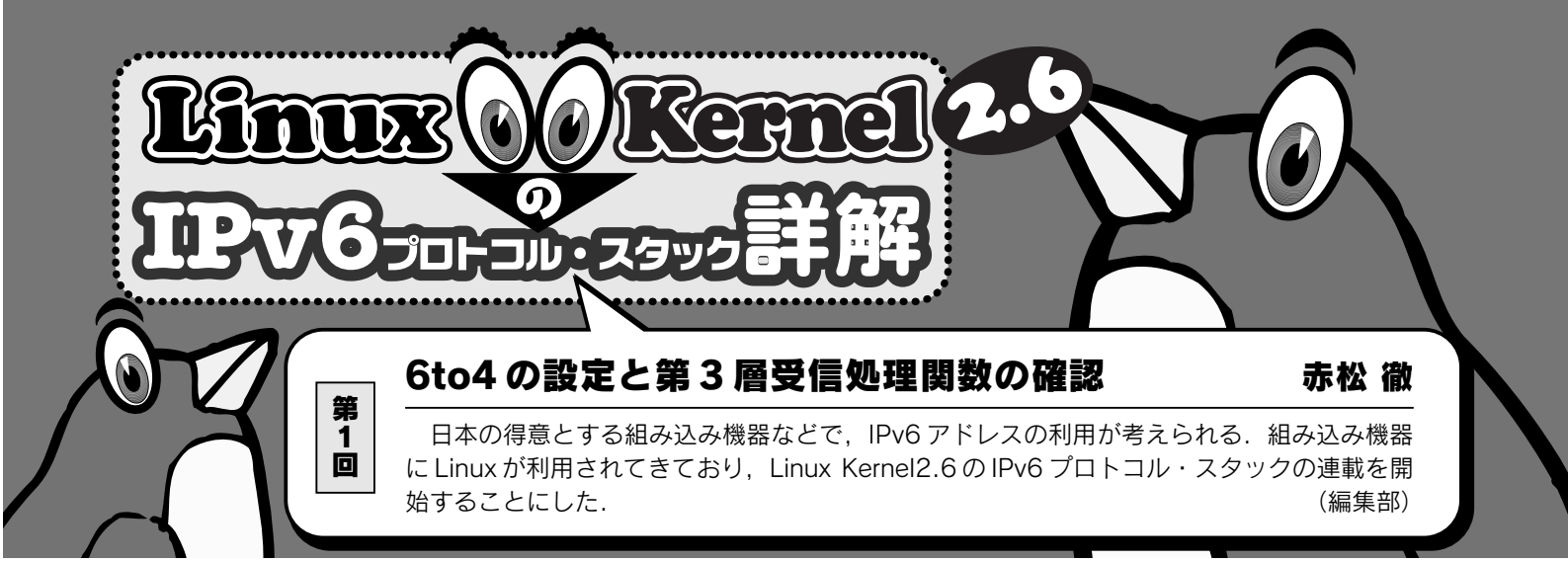

第1回

6to4の設定と第3層受信処理関数の確認

赤松 徹

日本の得意とする組み込み機器などで，IPv6アドレスの利用が考えられる．組み込み機器にLinuxが利用されてきており，Linux Kernel2.6のIPv6プロトコル・スタックの連載を開始することにした．（編集部）

IPv6の現状と組み込み機器への採用

筆者がIPv6を最初に使用したのは今から5年程前で，Linux Kernel2.4にパッチを当てて動作を確認しました．現在では，最新のKernel2.6にIPv6プロトコル・スタックがすでに組み込まれています．このLinux Kernel2.6のIPv6には，多くのUSAGI（UniverSAl playGround for Ipv6）チームが参加して開発に貢献しています．

アメリカやヨーロッパはIPv4アドレス（32ビットのIPアドレス）の割り当てを多くもっているので，いまだIPv6を積極的に利用していません．日本，中国，インドといったアジア圏では，元からIPv4の割り当てが少ないため，今後IPv6（128ビットのIPアドレスをもつインターネット・プロトコル）の積極的な利用が見込まれます．もちろん，プライベート・アドレスを利用することで，IPv6の実用が遅れていることも事実です．

近い将来，携帯電話やネット家電などでIPv6アドレスが利用されるようになることが考えられます．組み込み機器にLinux Kernelが利用されている現状も踏まえて，Linux Kernel2.6のIPv6プロトコル・スタックについて解説していきます．もちろん，IPv4処理も対称的に取り上げて説明します．

いきなりKernel内部の解説を始めても親しみがわきません．周りにあるLinuxマシンでIPv6の動作確認ができれば，「動くなら一度使ってみれば？」と協力的なご意見をいただけるものと思い，6to4[注1]を設定して，traceroute6[注2]から出発することにしました．UDP6[注3]とICMP6[注4]の両方を確認できる「一粒で2度おいしいtraceroute6」です．

注1：IPv4からIPv6に移行するときに，IPv6のデータ転送の単位をIPv4のデータ転送の単位にカプセル化して接続する手法．明示的なトンネル設定はしない．

注2：パケットの宛て先の装置までの経路情報を確認するコマンド．

注3：User Datagram Protocol．トランスポート層の通信プロトコル．

注4：Internet Control Message Protocol version 6．IPv6ネットワークで使用する通信プロトコル・ネットワークに異常事態が発生して，エラーが発生した場合，そのことを送信者に通知するIPv6のプロトコル．IPv4のICMPに加えて，近隣探索などの新しい機能を追加したIPv6用のICMPである．

6to4を設定してトンネリング処理

内部LANにIPv6を張り巡らせている環境も少ないでしょうし，公式のIPv6アドレスを取得している読者も多くないと思います．そこで，グローバルIPv4で接続しているLinuxマシンに6to4を設定してIPv6の実験を行います．デストリビューションはFedoraですが，別のものでも変わらないと思います．Kernelは最新版のKernel-2.6.15.2にアップデートしました．

それでは，最初にsit0（Simple Internet Transition）インターフェースを作成しましょう．設定コマンドは簡単です．新しいipコマンドでも可能ですが，使い慣れたifconfigコマンドで次のように設定しました．

```
# ifconfig sit0 up
```

次にグローバルIPv4アドレスからsit0に設定するIPv6アドレスを計算します．ここで60.56.229.25のIPv4アドレスを例にあげます．32ビットのIPv4アドレスの上位16ビットと下位16ビットを2組の4桁16進数に変換すると3c38:e519となります．さらに上位16ビットに2002:を付加すると，最終的に2002:3c38:e519::1/16というIPv6アドレスが求められます．このIPv6アドレスをifconfigコマンドでsit0に設定します．

```
# ifconfig sit0 add 2002:3c38:e519::1/16
```

トンネリング処理を一般公開しているkddilab.6to4.jpをゲートウェイに設定します．次のhostコマンドに-tでIPv4はA，IPv6ならクワッドA（AAAA）を指定して名前解決を確認してください．

```
# host -t A kddilab.6to4.jp
kddilab.6to4.jp. has address 192.26.91.178
# host -t AAAA kddilab.6to4.jp
kddilab.6to4.jp. has AAAA address
                                2002:c01a:5bb2::1
```

最後にkddilab.6to4.jpのアドレスをゲートウェイとしてrouteコマンドで設定すれば，トンネル越しにIPv6の世界が見えてきます（前方の::を入れるのを忘れてはいけない）．

```
# route -A inet6 add ::/1 dev sit0 gw
```


KEYWORD ── IPv6，Linux Kernel2.6，プロトコル・スタック，USAGI，6to4，traceroute6，UDP6，ICMP6，ifconfig，RFC894，ハッシュ法

```
0000 0000 0000 0000 0001 1010                    0001 1010 0000 1010 1000 1100 ①
0000 0000 0000 0000 0001 1010 0000 0000 0000 0000 0001 1010 0000 1010 1000 1100 ②
0000 0010 0000 0000 0000 0000 1111 1111 1111 1110 0000 0000 0000 0000 0000 0000 ③
0000 0010 0000 0000 0001 1010 1111 1111 1111 1110 0001 1010 0000 1010 1000 1100 ④
0    2    0    0    1    A    F    F    F    E    1    A    0    A    8    C    ⑤

fe8000000000000002001afffe1a0a8c 02 40 20 80     eth0
```

図1　MACアドレスからリンク・ローカル・アドレスを求める手順

```
# traceroute6 -n www.kame.net
traceroute to www.kame.net (2001:200:0:8002:203:47ff:fea5:3085) from 2002:3c38:e519::1, 30 hops max, 24 byte packets
 1  2002:c01a:5bb2::1  43.772 ms  24.035 ms  23.878 ms
 2  2001:200:0:1800::9c4:2  25.128 ms  24.066 ms  24.376 ms
 3  2001:200:0:1c04:230:13ff:feae:5b  27 ms  24.02 ms  24.382 ms
 4  2001:200:0:4800::7800:1  27.817 ms  25.404 ms  51.527 ms
 5  2001:200:0:8002:203:47ff:fea5:3085  18.389 ms  24.28 ms  24.761 ms
```

図2　traceroute6 で www.kame.net までの経路を確認する

図3　RFC894のフレーム・ヘッダ構造

```
                                        ::192.26.91.178
```

設定された情報をifconfigコマンドで確認しましょう．世界で唯一の公式IPv4アドレス(3c38:e519)を一部に取り入れて，グローバルIPv6アドレスとして識別します．プロバイダに固定IPで接続していなくても，グローバルIPv4アドレスを受け取れるマシンならば，この実験は可能です．

```
# ifconfig sit0 | grep Global
inet6 addr: 2002:3c38:e519::1/16
                                        Scope:Global
```

IPv6アドレスを調べてみる

ifconfigコマンドでIPv6アドレスを表示することができますが，ifconfigのソース・コードを読むとわかるように，仮想ファイルの/proc/net/if_inet6ファイル内容を読んで処理しています．したがって，次のcatコマンドでも確認できます．

```
# cat /proc/net/if_inet6
20023c38e5190000000000000000000001 04 10 00 80 sit0
00000000000000000000000000000001 01 80 10 80 lo
00000000000000000000000003c38e519 04 60 80 80 sit0
00000000000000000000000007f000001 04 60 90 80 sit0
fe8000000000000002001afffe1a0a8c 02 40 20 80 eth0
```

fe80/64で始まるIPv6はリンク・ローカル・アドレスです．ルータ内部の同じ回線上(同一リンク層)に接続しているIPv6マシン間限定で通信できるアドレスです．先程の2002:で始まるIPv6アドレスは公式IPv4アドレスを組み入れたものでしたが，fe80:で始まるリンク・ローカル・アドレスはネットワーク・カードが保持するMACアドレス(ハードウェア・アドレスとも呼ぶ)を組み入れたものです．

図1に示すように，MACアドレス(00:00:1A:1A:0A:8C)のネットワーク・カードに割り当てるリンク・ローカル・アドレスを求めましょう．

① MACアドレスを上位24ビットと下位24ビットに分割する
② 16ビット幅の分割した空間を入れる
③ 上位7ビットを1，分割した16ビット空間に0xfffeとするパターンを作成する
④ ②と③の論理和をとる
⑤ 16進数で表示した値が下位64ビットで，上位64ビットはfe80::と合成するとリンク・ローカル・アドレスとなる．

traceroute6でサーバの経路を確認する

有名なwww.kame.netにtraceroute6した実行結果を**図2**に示します．

経由するルータの数が極端に少ないことがわかります．その分IPv4より高速転送できるメリットがあります．

IPv6の世界に飛び込む最初のルータは，先程調べたkddilab.6to4.jpのアドレスであることも確認できます．

IPv6とIPv4を識別する

デバイス・ドライバはネットワーク回線を監視して，自分が取得しなければならないフレームを発見すると，そのフレームを取り込みます．RFC894(A Standard for the Transmission of IP Datagrams over Ethernet Networks. 1984.4)で定義されたフレーム・ヘッダ構造を**図3**に示します．このとき，パケット・タイプで示す2オクテット長(ネットワーク・バイト順)の