

FreeBSDを使ってIPv6を使い込む

Echigo BSD Users Group
2008/09/20(万代市民会館)
2008/09/28(update)
神保道夫

IPv4の世の中的情勢

- 2010年から2011年には枯渇する？
(地デジの2011年問題よりも早いのかも？)
- 逃げとして、キャリアグレードNAT(ISPがNATをしてIPv4アドレスを使いまわす)も考えられているが、最近の傾向として、1アプリケーションでTCPを200セッション程度張る物(ex. iTunes等)などもあり、現実的に稼働できるかどうかは未知数である。

では、IPv6にできるのか？

- まだ対応しているサイトが少ない
→IPv4に依存した世界が待っている
- IPv6に移行できるキラーアプリがない
ping6/traceroute6で遊んでみる?
Dancing Kameを見る?
踊るひろゆきを見る?
ipv6.2ch.netに書き込む?
branco TVを見る?
<http://flets-v6.jp/> とか?
- フレッツ網との共存はどうするのか？
- PPPoEでIPv6アドレスをやり取りする方法はない。(PPPではIPv6CPでIPv6アドレスが取得できるらしい・・・が、まだ実運用でやっているのはOCN IPv6だけ?)
- ネイティブIPv6に対応しているコンシューマ向けルーターはない。
(せいぜい、YAMAHAのRT58i位? 一般的なルーターは、フレッツ網のRAを垂れ流す、「IPv6ブリッジ機能」のみ)
- IPv6アドレスは普及するのか?
→課題は山積み

私のIPv6の歴史

- 2002年頃に、ADSLで常時接続+固定IPv4アドレス(/28)を貰って以来、IntecのIPv6線での接続を維持
(giftトンネル+RIPngによるルーティングという、かなり特殊な運用...)
- feel6のサービスが始まってからは、feel6のアドレスも取得
- WIDE ProjectのXCAST WGより、IPv6アドレスの割り当てを受け、EBUG内でのIPv6アドレスの管理を行う
- Bフレッツで閉域網の固定IPv6アドレスの配布が始まってからは、閉域網とインターネットとの接続のノウハウを調査
- かれこれ6年位IPv6とはお付き合い。

テストケース1-1

- Intec IPv6線(1)とfeel6(2)による疑似マルチホームの実験
(1)はgif0+RIPngによる接続
(2)は、gif1+dtcpによる接続
(1)をデフォルトルートにし、(2)のアドレスに関しては、(2)を通るようにする
- 手法
外向きのI/Fには、(1)と(2)のIPv6アドレスを振る
デフォルトルートは(1)を設定する
/usr/local/etc/dtcpclient_script.confに、
 feel6_tunif='gif1'
 feel6_static_routes='2001:3c0::/35,3ffe::/16'
と書いておく(このファイルを作らなければ、自動的にデフォルト
ルートを設定してしまうので、feel6のアドレスだけルートを作る。
3ffe::/16を書いているのは、<http://www.feel6.jp/> が、3ffe:: なの
で止む無く...)

テストケース1-2

- IPv6の起動は早い方がいいので、`/etc/netstart`の、`netstart_ipv6`の後に、`/etc/rc.d/dtcpclient start`を書いて、`dtcpclient`を立ち上げるようにする。しかし、`dtcpclient`は、接続先ホスト名をDNSで解決しようとするため、自前でDNSを立ち上げ、そちらを使っていると、`named`が立ち上がらないとI/Fがupしないようだ。なので、`/etc/hosts`に書いておくといいと思う。また、`dtcpclient`の起動順序を直す必要があるので、

```
# PROVIDE: dtcpclient
# REQUIRE: network_ipv6
# BEFORE: NETWORKING
```

と変える必要がある。
- なお、この方法は`mergemaster`をしてしまうと`/etc/rc.d/dtcpclient`が消えるので注意が必要かも。

テストケース1の結果(1)

- traceroute6を試みる(IPv6版traceroute。FreeBSDには標準で付いてくる)
% traceroute6 www.jp.freebsd.org
traceroute6 to www.jp.freebsd.org (2001:2f0:104:1:2e0:18ff:fea8:16f5) from 2001:200:xxx:1::1, 64 hops max, 12 byte packets
1 2001:200:500:1800:2d0:b7ff:fe26:127 31.250 ms 29.806 ms 31.873 ms
...
15 2001:2f0:104:1:2e0:18ff:fea8:16f5 40.944 ms 39.889 ms 41.984 ms

% traceroute6 www.feel6.jp
traceroute6 to www.FreeBit.NET (3ffe:8200:0:10:2d0:b7ff:fe91:20fb) from 2001:3e0:xxx:1::1, 64 hops max, 12 byte packets
1 2001:3e0:0:30:230:b6ff:fedf:541b 24.373 ms 19.760 ms 20.047 ms
...
4 3ffe:8200:0:10:2d0:b7ff:fe91:20fb 20.980 ms 20.943 ms 21.966 ms
- このように、それぞれのルートに応じて使うI/Fを使い分けてくれる。

テストケース1の結果(2)

- 一見よさそうだが...

```
route add 2001:0240::/32 -interface gif1
```

と、I/IJのルートをfeef6経由で設定してみると、

```
% traceroute6 www.iij.ad.jp
```

```
traceroute6 to www.iij.ad.jp (2001:240:bb42:b000::1:80) from
```

```
2001:200:xxx:1::1, 64 hops max, 12 byte packets
```

```
1 2001:3e0:0:30:230:b6ff:fedf:541b 31.114 ms 30.552 ms 29.901 ms
```

```
...
```

```
10 2001:240:bb42:b000::1:80 43.470 ms 34.503 ms 34.956 ms
```

- 到達できるものの、始点IPv6アドレスが、本来考えている、(2)のI/F アドレスではなく、(1)のアドレスになっている！
- IPv6アドレスは、src addr はデフォルトでは dest addr に対して最長一致で選択されます。つまり、2001:240:: のルートに対しては、2001:3e0::よりも2001:200::の方を優先して選択してしまいます
→IPv4とは違う！

default address selection(1)

- RFC3484に、default address selectionというのがあります。これは、このようなマルチホームでの運用時のsrc addr 及び dst addr の選択方法であり、FreeBSDにはこれが実装されているため、これを使います。
- /etc/ip6addrctl.confに、

```
::1/128          70 0
::/0             60 1
::ffff:0:0/96   50 1
2001:240::/32   40 2(←これを追加してみる)
2001:3e0::/35   30 2
3ffe::/16       20 2
2001:c90::/32   10 3 (←これは後で説明)
とりあえずこんな感じで書いてみる。その後、
# ipv6addrctl flush
#ipv6addrctl install /etc/ip6addrctl.conf
```

default address selection(2)

- 再度、www.iij.ad.jp にtracertoe6をしてみる。

```
% traceroute6 www.iij.ad.jp
traceroute6 to www.iij.ad.jp (2001:240:bb42:b000::1:80) from 2001:3e0:xxx:1::1,
64 hops max, 12 byte packets
 1 2001:3e0:0:30:230:b6ff:fedf:541b 22.395 ms 21.484 ms 21.983 ms
...
10 2001:240:bb42:b000::1:80 22.481 ms 22.558 ms 23.963 ms
```

- ちゃんと、gif1側のルートを使ってくれた。

テストケース2

- テストケース1の環境に、更にBフレッツの閉域網のIPv6アドレスを加え、ルーティングしてみる。
- NTT東日本の場合、Bフレッツの使用するIPv6アドレスは、2001:c90::/32なので、この範囲のアドレスをうまくルーティングしてやる。
- 私の自宅では、ひかり電話経由で閉域網IPv6アドレスが、RA(Router Advertization)経由で、/64で流れてくるので、まずそこからIPv6アドレスのネットワーク部と、デフォルトゲートウェイアドレスを求める。うちだと、IPv6ネットワークは、2001:c90:59ba:88f9::/64で、デフォルトゲートウェイがfe80::214:f1ff:fe57:81aなので、まずサーバーに適当にIPv6アドレスを振り、
route add -inet6 2001:c90::/32 fe80::214:f1ff:fe57:81a
を書き、ip6addrctl.confを書く(優先度を一番低くすることがポイント)
- ただし、ひかり電話は、勝手にTeredo Addressも設定してくれるので、ルーターのDHCPをOFFにして、FreeBSDでDHCPサーバーをやらせた方が良い。

テストケース2の結果(1)

- このグローバルネットワークに、そのままWindows PC(ここでは、Vista SP1を想定)を繋いでしまうと、FreeBSDサーバーでrtadvdを動かしている場合、グローバルのIPv6と閉域網のIPv6アドレスが付く。しかし、デフォルトゲートウェイが、グローバルと閉域網の2つ付いてしまうので、コマンドプロンプト上でRAを禁止してみます。

```
C:¥>netstat -rn
```

```
...
```

```
IPv6 ルート テーブル
```

```
=====
```

```
アクティブ ルート:
```

```
  If メトリック ネットワーク宛先   ゲートウェイ
```

```
  10  276  ::/0                fe80::214:f1ff:fe57:81a(この、If番号を覚えてください)
```

```
C:¥>netsh
```

```
netsh>interface ipv6
```

```
netsh interface ipv6>set interface 10 routerdiscovery=disabled store=persistent  
(10は、先ほどのIf番号)
```

- これで、永久的にRAは聞かなくなります。
- 別の解として、default routeにpreferenceを設定する方法もあるそうです。

テストケース2の結果(2)

- ところが、RAを聞かなくなったら、勝手にTeredo Addressを設定して、デフォルトルートを作ってくれました！更に、6to4とかisatapとか、訳のわからないものが邪魔しそうだったので、全部OFFにしちゃいます^^;

```
C:¥>netsh
```

```
netsh>interface 6to4
```

```
netsh interface 6to4>set state state=disabled undoonstop=disabled
```

```
C:¥>netsh
```

```
netsh>interface isatap
```

```
netsh interface isatap>set state disabled
```

```
C:¥>netsh
```

```
netsh>interface ipv6
```

```
netsh interface ipv6>teredo disable
```

テストケース2の結果(3)

- この後、Windowsのネットワークの設定で、手動でIPv6アドレスを振り、デフォルトゲートウェイ、DNSサーバーをFreeBSDのサーバーに振れば、見事に通信できました！ただし、<http://flets-v6.jp/> は、FreeBSDサーバー経由だとなんかおかしいので、更に次のコマンドを入れることにより、サクサクになりました。

```
C:¥>route add 2001:c90::/32 fe80::214:f1ff:fe57:81a -p
```

- これは、2001:c90::/32に対して、再起動時も常にルートを設定する(-p)方法です。
- なお、この設定は、Windowsの自動設定を殺しますので、ある意味危険です。Windows Vistaには、IPv6匿名アドレス機能、RFC3041「Privacy Extensions for Address Configuration in IPv6」というのがありますが、固定IPになってしまうので、セキュリティの面を考えると、考える必要がありそうです。

テストケース3(1)

- なんとか、IPv4はNAT経由で、IPv6はグローバルアドレスをつけ、なおかつ<http://flets-v6.jp/>が見られないものか...
- ということで、pf+bridgeで実験してみました。

テストケース3(2)

- FreeBSDの設定
 - cloned_interface="gif0 gif1 bridge0"
 - autobridge_interfaces="bridge0"
 - autobridge_bridge0="re0 re1"
 - ifconfig_bridge0="up"
 - ifconfig_re1のIPv6の設定は削除
- Windows側の設定
 - 設定は、前述と同じ

テストケース3(4)

- 問題が山ほど発生！
 - ・サーバーへのIPv6アクセスが、bridge経由だとなぜかできない。サーバーからping6を打つと、その間だけ開通。
→なぜか、re1側にre0と同じセグメントのIPv6アドレスを振り、Windows側をデフォルトルートにすると解決
 - ・prefixpolicyが効かない。サーバーのdefault routeから抜けていくため、ソースアドレスフィルタをされたらアウト。
 - ・複数IPv6アドレスを振ると、勝手に最短一致のアドレスをデフォルトのIPv6アドレスにしてしまう。従って、IPv6アドレスとして、2001:200:xxx と2001:3e0:yyy のアドレスを使って、2001:3e0:yyy のアドレスを優先にすることはできないのか？
- 結論：bridge経由のIPv6使用はまだ難あり

今後の課題

- テストケース4の詳細テストを引き続き行いたい(時間あるかな・・・)
- RFC3041関係の調査(netshコマンドの詳細な調査)
- Microsoft関係でデフォルトで実装されているIPv6関係の調査
- 等々

まとめ・参考サイト

- IPv4の枯渇問題に対して、IPv6を使った解決方法は、まだまだ敷居が高すぎる。特に日本では、NTT問題があるためにさらに厄介。早急な対策を考える必要だ。
- 参考サイト
 - <http://www.nic.ad.jp/ja/materials/iw/2006/proceedings/T8-2.pdf>
 - <http://www.imasy.or.jp/~ume/presentation/CBUG-20031129/>
 - <http://www.tumori.nu/IPv6/addrselect.html>
- Special Thanks to norkさん、umeさん

netsh コマンドのヘルプ

おまけ

- FreeBSD標準のtraceroute/traceroute6で、AS pathを見たくありませんか? portsには、nanog-treacerouteとか、いろいろありますが、portsはやだとか、オプション指定がめんどくさいって人のために(それは私だ(^^;)

http://karl0204.at.webry.info/200809/article_8.html

http://karl0204.at.webry.info/200809/article_7.html

でパッチを公開しています。この他、私のブログ(<http://karl0204.at.webry.info/>)では、いろいろなサーバー構築テクニックを公開していますので見てください。