データセンタ、サーバ構築における IPv6トラブルシューティング

株式会社ビーコンエヌシー
國武　功一

2013年11月27日

Agenda

- よくあるパターン
  - ＤＮＳ周り
  - ICMPv6 (Path MTU Discovery)
  - 設定
  - アドレスつけたのに！
  - ヒットしません
  - その他

※1すべてフィクションです、わりと本当に

※2 IPv6 only環境話は取り上げません

DNS周り

- なんか遅い、その１
  - 支店からウェブアクセスすると早い。本店からアクセスすると、妙に遅い。
- なんか遅い、その２
  - 外部API叩いている機能使うと重い
- なんか遅い、その３
  - よくわかんないけど、重い
- ある日突然タイムアウトの嵐

なんか重い、その1

- 事象
  - 支店からウェブアクセスすると早い。本店からアクセスすると、妙に遅い。
- 原因
  - サーバの再構築後、IPv6アドレスの付与を忘れ、AAAAを残したままだった（フォールバック問題）
  - 支店にはIPv4環境しかなく、本店には、IPv4/IPv6の接続性があり、IPv6から、IPv4へのフォールバックが発生していた。

IPv6アドレスの付与漏れ

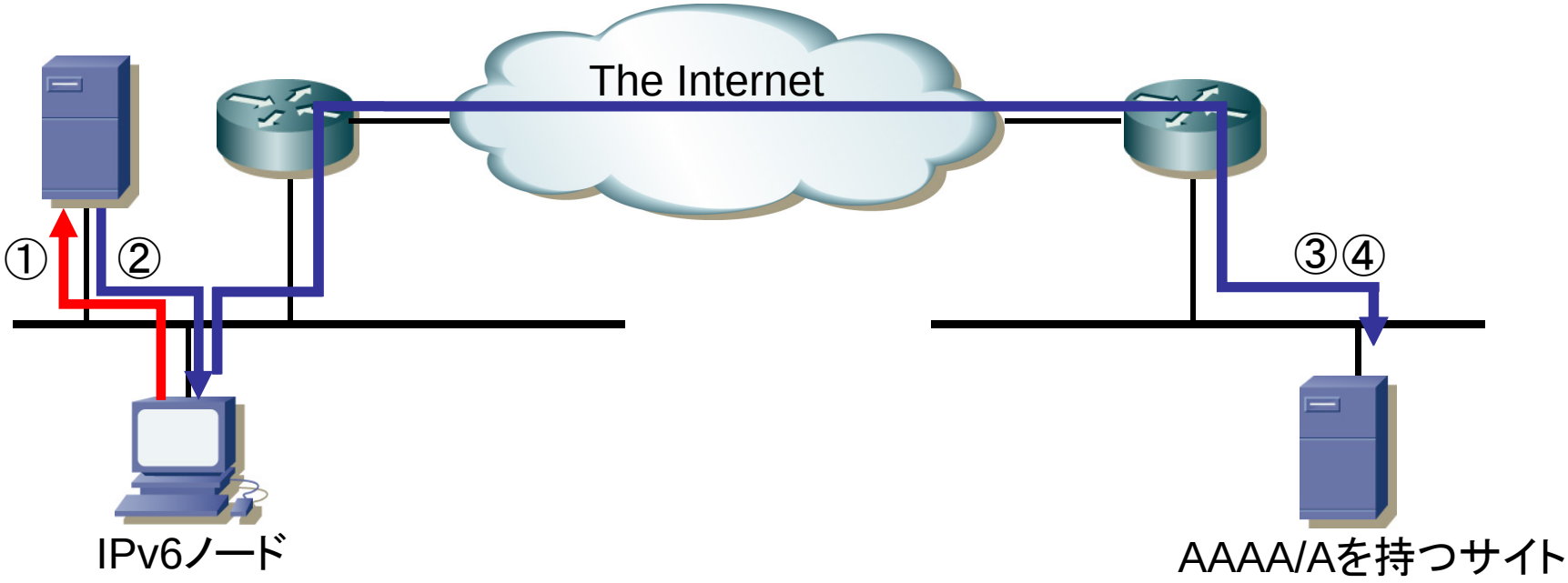
```
;; Server
www        IN     A          192.0.2.1
           IN     AAAA       2001:db8::80
```

移行前にはついていたアドレス

そもそもIPv6アドレスに対して、監視がなされていなかったのも問題

そして IPv6 -> IPv4のフォールバックが

①アドレスを引く
②AAAA RR, A RRが返る
③IPv6で接続
④IPv6で接続できないと、IPv4へフォールバック



なんか遅い、その2

- 事象
  - クラウドのAPI叩いている機能を使うと重い
  - sshでログインする時、妙なひっかかりがある
- 原因
  - Glibc2.6以降を使っている場合のLinuxのリゾルバの挙動と、Firewallとの、華麗な共演

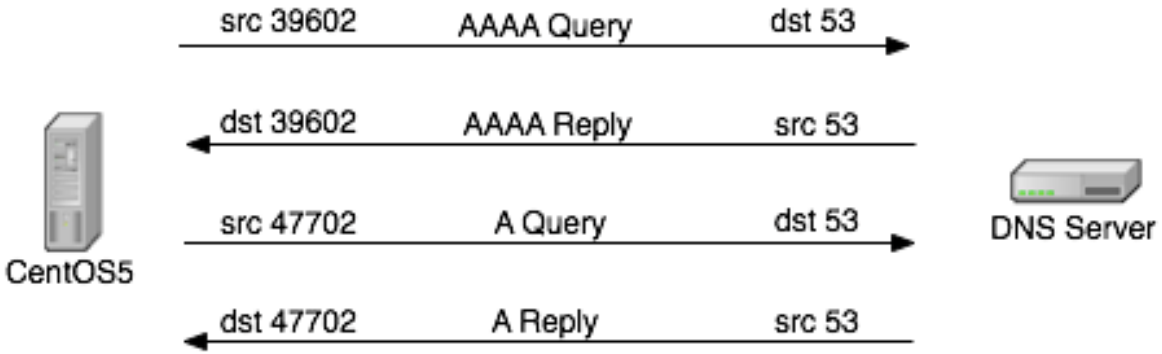
DNSクエリに関するOSの対応

- クエリ順序はOSで異なる
  - AAAAクエリを先に実施するOS
    - Windows XP、Linux
  - Aクエリを先に実施するOS
    - Windows Vista、Windows 7、FreeBSD、Mac OS X

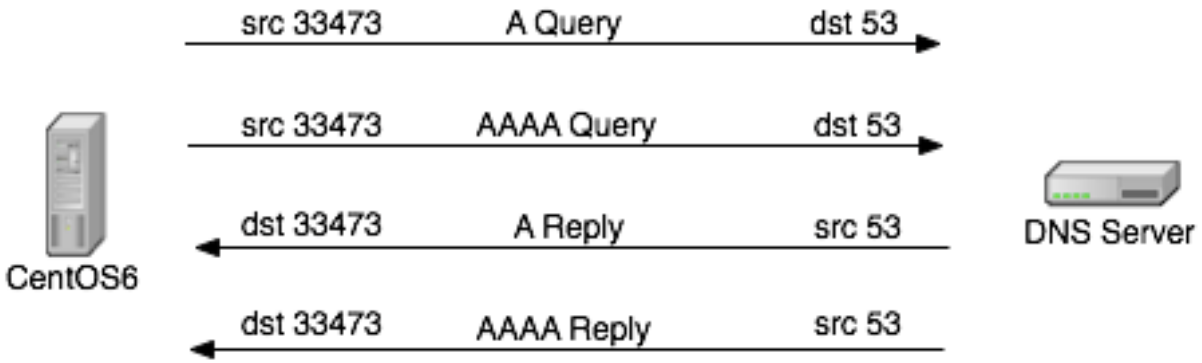
InternetWeek 2010 北口氏資料より一部抜粋 (p.64)
http://www.nic.ad.jp/ja/materials/iw/2010/proceedings/s2/iw2010-s2-01.pdf

が、Glibcのバージョンによって…

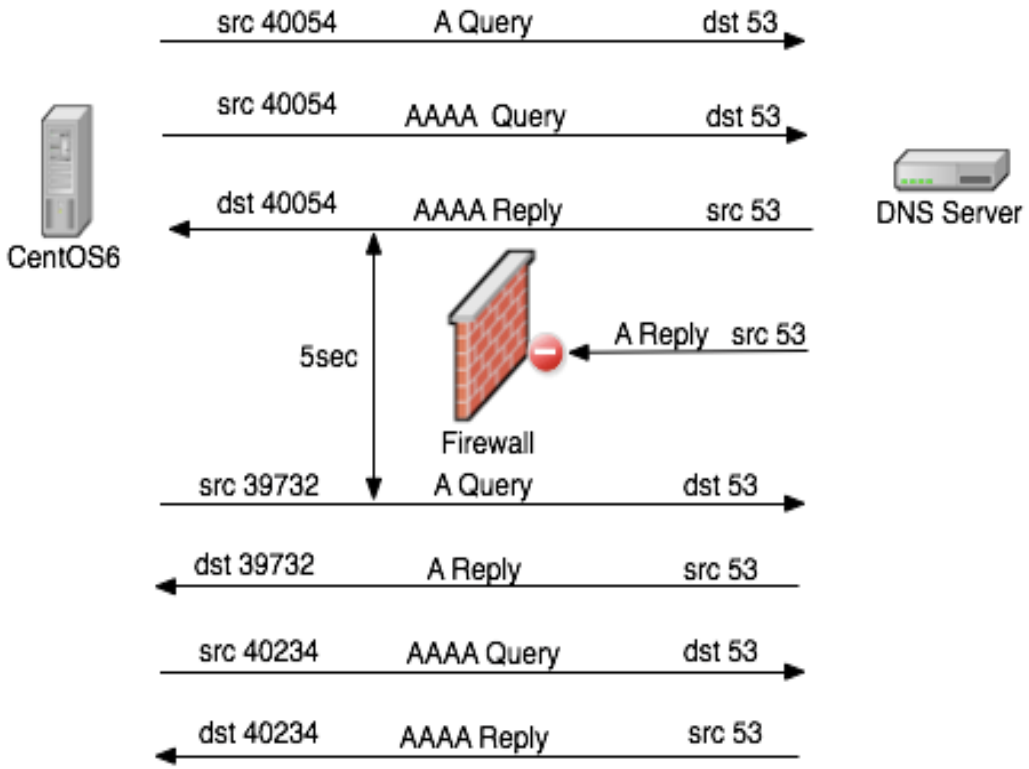
- RHEL5/CentOS5



- RHEL6/CentOS6 and so on.



挙動が変わって万歳！？



一部のファイアウォールの実装では、同一ポートからのクエリを同一のセッションとみなし、結果返信が落とされてしまうものがある

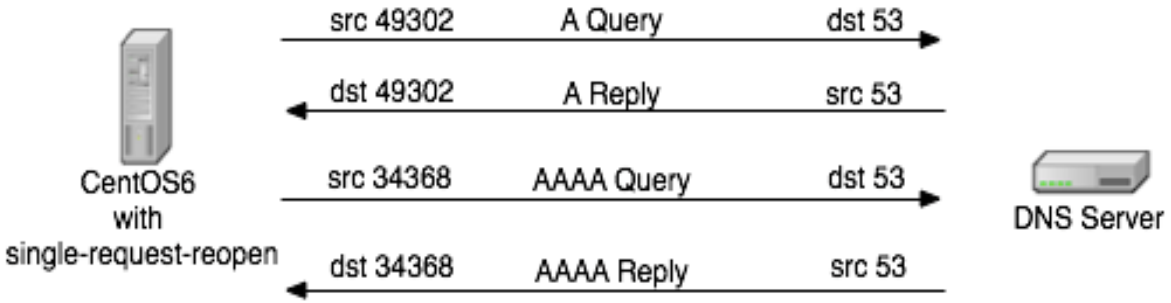
この問題の罪なところ

- 名前は最終的に引ける
- 若干遅いぐらい(標準設定で5秒でfallback)
  - options timeout:1 なら、もっと短い(そして発覚しずらい)
- 最近のサーバはAPI連携で、DNSを引くことも
  - 普通にクライアントとして使われる場合には、DNSの結果はキャッシュされないこともあり、ユーザの1リクエストに対して、複数回APIを叩くと……

single-request-reopenを設定する

- /etc/resolv.conf にオプションとして設定すると、クエリ毎にポートを変えるようになる(socketを作り直す)

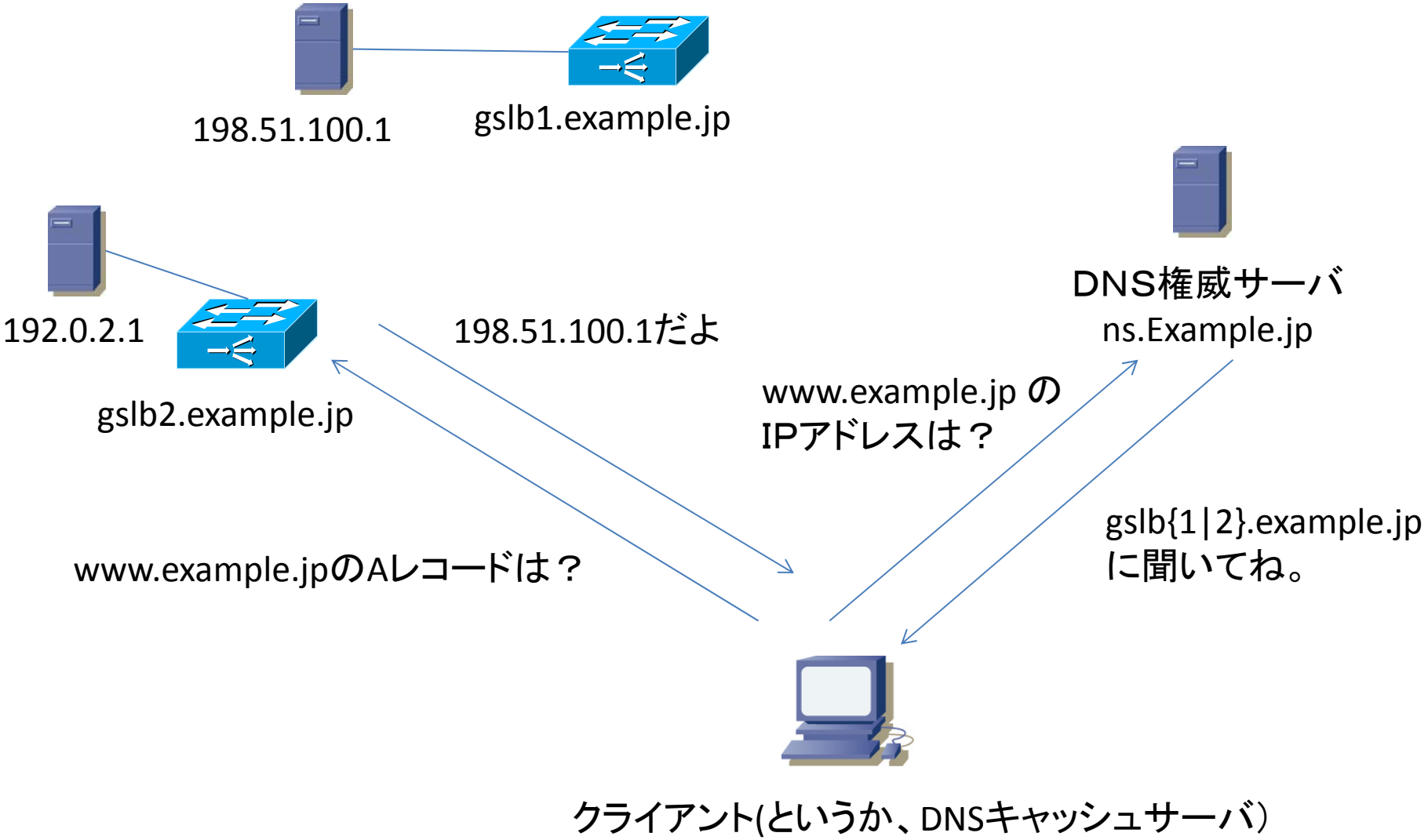
```
search example.jp
nameserver 2001:db8:0001::53
nameserver 2001:db8:fffff::53
options single-request-reopen
```



なんか遅い、その3

- 事象
  - よくわかんないけど、重い
- 原因
  - ある事例では、GSLBなどが、AAAAに応答せず、タイムアウトすることで、AAAAのQueryを投げるクライアントからのアクセスが結果的に遅くなる。
  - 導入前に、Aレコードなどしか利用を想定していない、もしくはテストをしていない。

GSLBのざっくりとした仕組み



ある日突然タイムアウトの嵐

- 事象
  - ある日、ULAを使っているネットワークで、突然タイムアウトの嵐。
- 原因
  - ULAに関する逆引きリクエストが Locally Serverd DNS Zonesの設定漏れで、IANA管理のサーバなどに聞きに行っていた。これが、IANA管理のDNS権威サーバの障害などで、タイムアウトを起こし障害へ発展。
  - Locally Served DNS Zones設定漏れに起因する障害（RFC6303)

DNS関連（Summary)

- 設定不備がほとんど
  - DNS権威サーバおよびDNSキャッシュサーバに対する知識の欠如
    - メーカー側、ユーザ側
  - IPv6サービスを提供していることが共有されない、またされ続けない。
  - IPv6でのサービスレベルが、IPv4のものと比べて、低くなってしまっている（積み重なっている運用経験が、活かされない）

ICMPv6 (PMTU問題）

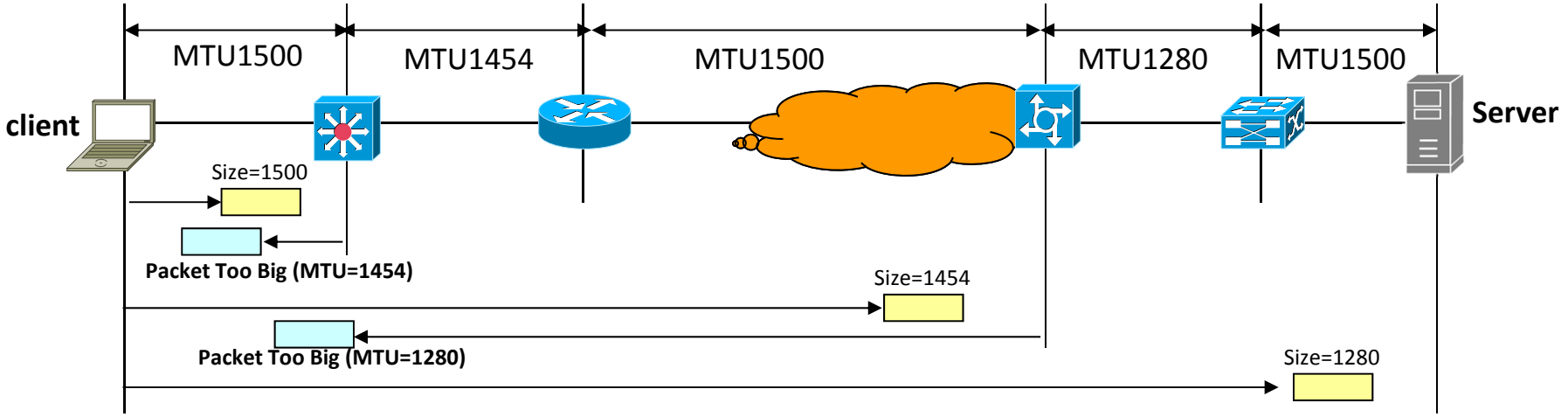
- アクセスできたり、できなかったり

アクセスできたり、できなかったり

- 事象
  - なんかアクセスできない
  - telnet すると、port は反応する。
- 原因
  - Path MTU Discoveryの動作に必要なICMPv6がフィルタされて、PMTUを越えるパケットが通らない(コンテンツのサイズによって、アクセス可、不可が変わる）

Path MTU Discovery

- IPv6 では中継ノードでフラグメントしない（始点ノードが実施）
  - IPv4 ではルータ等の中継ノードがフラグメントを実施
  - 送信パケットに対する ICMPv6 Error Message を受信時、MTU を変更
    - 最初のリンクのMTU が初期値
    - ICMPv6 Packet Too Big Message 受信時、始点ノードでフラグメントして再送
  - IPv6最小MTU は、1280byte
    - L2 SWのMTUにひっかかった場合は破棄される
    - Path MTU Discovery の実装が難しいノードは 1280byte 固定（実質難しい）



設定

- どっちで書くの？
- Typo ？
- 挙動が違う...

どっちで書くの？

- IPv6アドレスを設定ファイルに記載する際に、[]でアドレスをくくる場合と、不要な場合とが同一アプリ内でも、分かれている。

Postfixの場合

- []で囲む必要があるもの
  - mynetworksやdebug_peer_listのように、Postfixマッチリストを設定する場合、“type:table”形式と混乱しないためにも、IPv6アドレスは、[]で囲う必要があります

```
# mynetworks = hash:/etc/postfix/network_table
mynetworks = 127.0.0.0/8 [::1]/128
```

Typo?

- ApacheでのACL
  - 2001:db8:0:1000/64とはできないので注意。きちんとネットワークアドレスを指定してやる必要がある。もし誤って記載しても、エラーなどは出ないので注意

```
AuthName “Staff Only”
AuthType Basic
AuthUserFile “/var/www/www.example.jp/.htpasswd”
Require valid-user
Order Deny,Allow
Deny from all
Allow from 192.0.2.1
Allow from 2001:db8:0:1000::/64
Satisfy Any
```

Typo??

F.A.E.B.D.A.E.D.F.F.C.F.C.3.2.4.0.0.0.1.0.0.0.0.8.B.D.0.1.0.0.2.IP6.ARPA

F.A.E.B.D.A.E.D.F.F.C.F.C.3.2.4.0.0.1.0.0.0.0.8.B.D.0.1.0.0.2.IP6.ARPA

F.A.E.B.D.A.E.D.F.F.C.F.C.3.2.4.0.0.0.0.1.0.0.0.0.8.B.D.0.1.0.0.2.IP6.ARPA

逆引き設定のtypo防止のため、自動生成がお勧め

$ arpaname 2001:db8:0:1000:423c:fcff:dead:beaf
F.A.E.B.D.A.E.D.F.F.C.F.C.3.2.4.0.0.0.1.0.0.0.0.8.B.D.0.1.0.0.2.IP6.ARPA

Typo???

 # ip route add 2001:db8::/64 via ¥

2001:db8:ffff::cafe/64 dev eth0

Error: an inet address is expected rather than "2001:db8:ffff::cafe/64".

挙動が違う…その１

```
# 例 (IPv4のみの設定)
< VirtualHost 192.0.2.80:80 >
    ServerName www.example.co.jp
  …
</VirtualHost>

# 例 (IPv6のみの設定)
#<VirtualHost *:80>
<VirtualHost [2001:db8:0:1000::80]:80>
    ServerName www.example.co.jp
  …
</VirtualHost>
```

設定変更するときに、とりあえずIPv4側だけに設定をいれて、IPv6側への反映を忘れる…

挙動が違う…その1(Cont)

```
# 例 (IPv4,IPv6の両方の接続を受け入れる)
<VirtualHost 192.0.2.80:80 [2001:db8:0:1000::80]:80>
    ServerName www.example.co.jp
  …
</VirtualHost>
```

特に理由がないなら、一緒にしてしまうのも手（特にSSL設定の場合等）

挙動が違う...その2

```
# ip6tables -A HOST_FILTER-INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

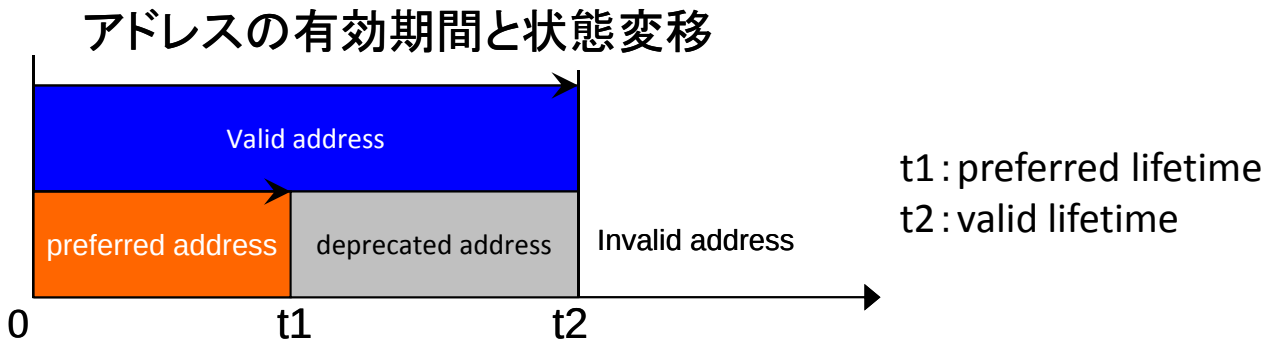
設定は入るのに、期待した挙動をにならない！
RHEL5/CentOS5では、ip6tablesの connection trackingはサポートされません。
まともに使うなら、RHEL6以降を使いましょう

アドレスつけたのに！

- 事象
  - ifconfigでアドレスが付いていることを確認したのに、そのアドレスを使って通信できない
- 原因
  - 実際には、DADが働いていて、tentative addressの状態になっている。

IPv6アドレスの状態、アドレスのlifetime

- tentative address
  - インタフェースに付与されていないアドレスでNDメッセージにしか使用できない。この時点でアドレスの一意性をDADで確認する。
- preferred address
  - インタフェースに付与されたアドレス。アドレスが一意で通信可能な状態
- deprecated address
  - 有効ではあるが、新規通信への使用をしないことが望まれる
- valid address
  - Preferredとdeprecatedのアドレスの双方を指す
- Invalid address
  - 有効アドレスの有効期間が過ぎるとこの無効アドレスになる



ナウでヤングな若者が使うのはipコマンド

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
   link/ether 00:21:aa:bb:cc:dd brd ff:ff:ff:ff:ff:ff
   inet 192.0.2.147/24 brd 192.0.2.255 scope global eth0
   inet6 2001:db8:0:1000:423c:fcff:dead:beaf/128 scope global tentative dadfailed
      valid_lft forever preferred_lft forever
   inet6 2001:db8:0:1000:3875:e798:dead:cafe/64 scope global secondary dynamic
      valid_lft 594645sec preferred_lft 75645sec
```

```
eth0      Link encap:Ethernet  HWaddr 00:21:aa:bb:cc:dd
          inet addr:192.0.2.147  Bcast:192.0.2.255  Mask:255.255.255.0
          inet6 addr: 2001:db8:0:1000:423c:fcff:dead:beaf/64 Scope:Global
          inet6 addr: 2001:db8:0:1000:3875:e798:dead:cafe/64 Scope:Global
```

Ifconfigコマンドでは、IPv6アドレスの状態、
ライフタイムなどがわからない

ヒットしない

- 事象
  - 調査依頼が来たが、該当するIPアドレスのログがない。
- 原因
  - Syslogなどの出力はテキストであるため、IPv6の省略表記によっては、grep や検索システム等で、ヒットしないことがある。

IPv6アドレス表記

- 128bit を 16bit 毎に 8分割後、各フィールドを 16進数表記にして、“ : ”(コロン)で区切る。

  00100000000000010000110110111000000000000000000000000000000000000000000010000001100010100111111111111111100001111001001000001011 10

  ↓

  2001:0db8:0000:0000:0206:29ff:fe1e:482e

- 先行する 0 は省略可能。但し、各フィールドには少なくとも 1つの数値を含むこと。

  2001:0db8:0000:0000:0206:29ff:fe1e:482e

  ↓

  2001:db8:0:0:206:29ff:fe1e:482e

- 16bit の 0 または、16bit の 0 が複数連続するフィールドを 1箇所のみ、:: を用いて省略する。

  2001:db8:0:0:206:29ff:fe1e:482e

  ↓

  2001:db8::206:29ff:fe1e:482e

省略記法について、詳細は [RFC5952] (A Recommendation for IPv6 Address Text Representation)を参照

では以下の場合は？

- 2001:0db8:0000:0000:fff0:0000:0000:000f
  ＝＞　2001:db8::fff0:0:0:f

・ダメな例

2001:db8:0:0:fff0::f

2001:db8::fff0::f

現状、一般的にRFC5952が守られているかどうかは不明（過渡期…）

トラブルシュートの友

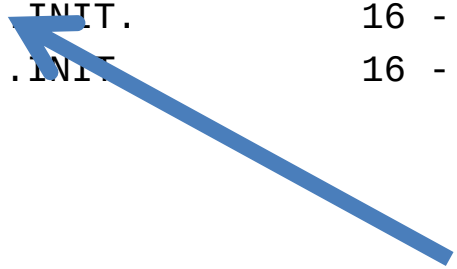
- 扱うプロトコルに対する基本的な知識
- dig +norecは友達
- ipコマンドも友達
- tcpdumpコマンドも友達（wiresharkがきっと手助けしてくれるはず）
- 切り分けのための環境を持っておこう（Firewallが存在しない環境、IPv4　only環境）
- テストテストテスト！！

あれ？いままでとあまり変わらないような……気もする

その他

- ssコマンドも友達
  - コマンドが、IPv4を文字列とした固定長を想定していて、アドレスが切り捨てられてしまう(netstatなど)
  - ntpqなど、代替コマンドがないものも(だれかご存じなら教えてください)

```
$ ntpq -pn
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
+210.173.160.27  172.29.3.60      2 u  685 1024  377    4.022   -0.222   0.382
+210.173.160.57  172.29.3.60      2 u  172 1024  377    3.889   -0.175   0.329
*210.173.160.87  172.29.3.50      2 u  889 1024  377    3.803   -0.084   0.169
 2001:3a0:0:2001 .INIT.          16 -    - 1024    0    0.000    0.000   0.000
 2001:3a0:0:2005 .INIT.          16 -    - 1024    0    0.000    0.000   0.000
 2001:3a0:0:2006 .INIT.          16 -    - 1024    0    0.000    0.000   0.000
```



すべてのアドレスが表示されず、切り捨てられている