IPv6 への移行に関する
データセンター事業者リファレンスモデル

IPv6 普及・高度化促進協議会
ビジネステストベッド WG
2011 年 6 月

1. データセンター事業者のリファレンスモデル

1.1. *前提となるネットワーク*

データセンター事業者リファレンスモデルの前提となるネットワークを図 1 に示す。

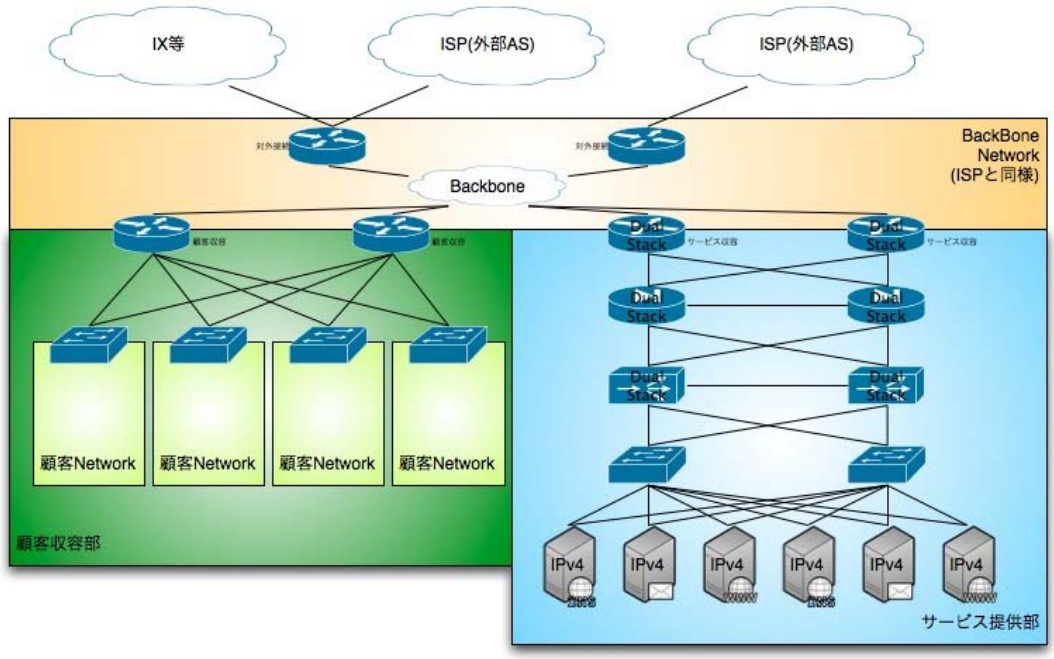

図 1　データセンター事業者リファレンスモデル：前提ネットワーク

一般に、データセンター事業者は機器設置場所の提供、インターネットに対して接続するためのネットワークの提供、及び事業者によっては VPS 等を提供している。

このうち、インターネットに対して接続する為のネットワークは一般にバックボーンと呼ばれ、この部分は ISP と同様のネットワークとなっている場合が多い。また、顧客収容部に関しては、本質的にはスペースとネットワークの提供であり、これも ISP の形式と差はない。

これに対しサービス提供部は、いわゆるサーバの集合である。

サービス提供部にて提供されるサービスには様々なものが考えられるが、代表的なものとして DNS、メール、Web がある。本モデルでは、提供サービスとして、この 3 種類のサービスが提供されているものを取り扱う。

本モデルにおいては、以下のことを前提条件とする。

- バックボーンネットワークは、既に IPv4/IPv6 の Dual Stack 化が完了していること

- 各ネットワークデバイスは、既に IPv4/IPv6 の Dual Stack 化が完了していること
- サーバ類は UNIX 系であること。
- サーバアプリケーションはフリーウェアを利用していること
  - 商用のサーバアプリケーションを利用している場合、アプリケーション自身が IPv6 に対応していること

1.2. *新たに必要となるもの*

本ネットワークにおけるサービス提供部に関して、新たに必要となるものは存在しない。

但し、商用のサーバアプリケーションを利用している場合、サーバアプリケーションが IPv6 に対応している必要があるため、購入する必湯がある場合がある。

1.3. *移行手順*

サービス提供部の移行に関しては以下のような手順を採ることになると考えられる。

1. IPv6 対応に関する調査
   (ア) サービス提供部で利用している各種アプリケーションの IPv6 対応状況調査
   (イ) サーバ OS 自身の IPv6 対応状況調査
   これらの調査を行い、サービス利用者が IPv6 で接続してきた場合に問題となり得る部分がないことを確認する。
2. サーバ OS の IPv6 対応設定の投入
   (ア) （必要であれば）サーバ OS のバージョンアップの実施
   (イ) サーバに対する各種設定の投入（9.4.1.3 参照）
   (ウ) （必要であれば）サーバの再起動の実施
3. 各種サーバアプリケーションの IPv6 対応設定投入
   (ア) DNS サーバの IPv6 対応（9.4.1.4 参照）
      ① （必要があれば）DNS サーバアプリケーションの更新
      ② DNS サーバの IPv6 対応設定投入
      ③ DNS サーバの再起動
   (イ) メールサーバの IPv6 対応（9.4.1.5 参照）
      ① （必要があれば）メールサーバアプリケーションの更新
      ② メールサーバの IPv6 対応設定投入
      ③ メールサーバ再起動
   (ウ) Web サーバの IPv6 対応（9.4.1.6 参照）
      ① （必要があれば）Web サーバアプリケーションの更新

② Web サーバの IPv6 対応設定投入

③ Web サーバ再起動

全体の手順書は付録に示す。

4. サービス試験

(ア) DNS を用いて名前解決を行えるかの確認

(イ) 各種サーバアプリケーションに対して、IPv4、IPv6 それぞれの環境から接続を試み、サービスが提供されていることを確認する

(ウ) 各種サーバアプリケーションの出力するログを確認し、それぞれ正しく、IPv4/IPv6 で通信できていることを確認する

1.4. *移行後のネットワーク*

データセンター事業者の移行後のネットワークを図 2 に示す。

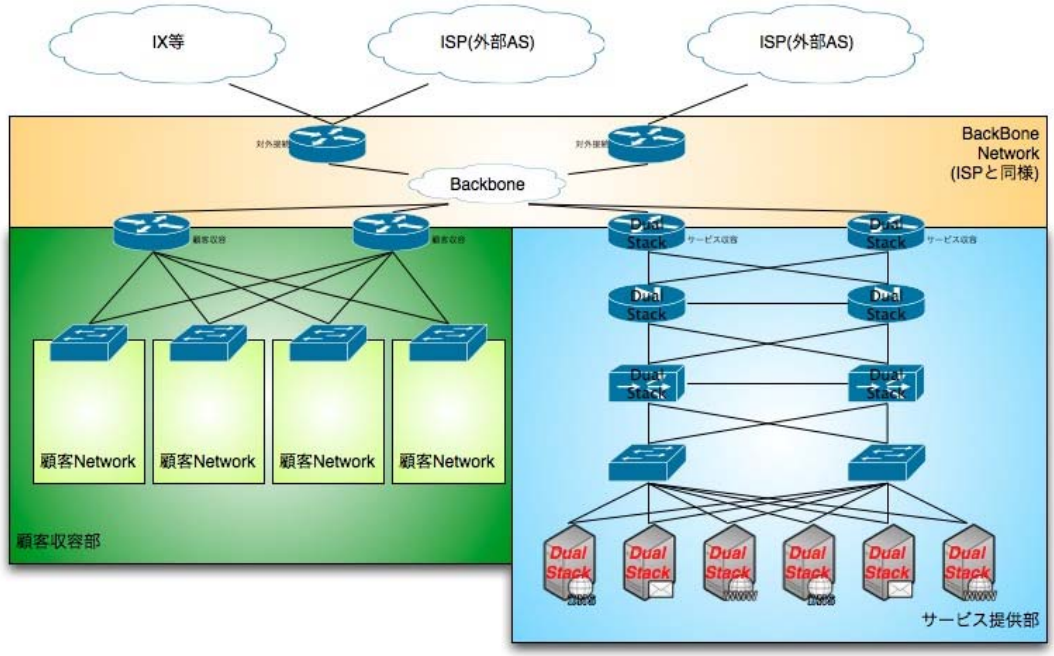

図 2 データセンター事業者移行後のネットワーク

1.5. *予想されるコスト*

一般に、ある種の特化したサーバアプリケーションや CGI を利用していない限り、サービスの IPv6 化に必要なコストは「調査・設定にかかる人件費」のみとなる。

表 1-1 データセンター事業者の予想移行コスト

大項目	作業項目	人数 （単位：人）	日数 （単位：日）	備考
現状調査	現状のサーバの状態とアプリケーションの状態の確認			
	使用機材、使用 OS のリストアップ	1	2	サーバ台数に依存。10 台程度を想定
	サーバ毎の利用アプリケーションのリストアップ	1	2	サーバ 10 台程度を想定
	IPv6 対応状況の確認	1	4	利用しているアプリケーション数に依存
IPv6 アドレスの取得	IPv6 アドレスを取得する。			
	JPNIC へ申請	1	1	
	JPNIC から割り当てを受ける	1		申請から割り当てまで 3 週間程度
ネットワークデザイン	dual stack 対応のネットワークを設計する			
	ネットワーク情報の取りまとめ、確認	1	1	
	ネットワーク構成の検討	1	2	VRRP 等を考慮
構成確定	設計に基づいて、アプリケーションのバージョン等を決める			
	OS 選定	1	1	
	アプリケーション選定	1	2	利用しているアプリケーション数に依存
詳細設計	OS、アプリケーションの設定の検討			
	IP アドレス設計	1	1	サーバ台数に依存。10 台程度を想定
	OS 設定設計	1	2	サーバ台数に依存。10 台程度を想定
	アプリケーション設定設計	2	4	提供しているサービスに依存
移行手順書作成	ネットワークデザイン、詳細設計に基づき、移行手順書を作成する。			
	詳細移行手順書の作成	2	4	利用しているアプリケーション数に依存
	作成した手順書の確認、机上シミュレート	2	2	提供しているサービスに依存
設定変更	各設定の変更を行う			

		OS の設定変更・確認	2	2	サーバ台数に依存。10 台程度を想定
		アプリケーションの更新	2	4	利用しているアプリケーション数に依存
		アプリケーションの設定変更・確認	2	4	利用しているアプリケーション数に依存

1.6. *注意点*

サービスを IPv6 に対応する場合には、以下の点に注意する必要がある。

- CGI 等のような付加プログラムを利用している場合、クッキーの取扱いや、通信時の IP アドレスの取扱いが問題ないことを確認する必要がある。
- SingleSignOn(SSO)を利用している場合、認証情報がどのように連携されるかを確認する必要がある。特に、IPv4 のみのサービスや IPv6 のみのサービスを混在させて SSO で認証管理を行っている場合、正しく認証情報が伝わらない可能性がある。
- IPv4 と IPv6 で制御が異なる機能が存在する。例えば、Linux におけるファイアウォールの実装は、IPv4 と IPv6 で異なるものとなっている。IPv4 用のファイアウォール設定コマンドは iptables だが、IPv6 用のファイアウォール設定コマンドは ip6tables である。従って、IPv4 では正しく設定されていても、IPv6 においては設定されていない場合がある。

サービスを IPv6 に対応させるには、上記の点に特に注意を払う必要がある。

付録：データセンター事業者移行手順書

平成 23 年 6 月

データセンター事業者移行手順書

本手順書で扱うサーバの OS は CentOS5 とする。

基本ソフトの **IPv6** 化

DNS について

DNS サーバの IPv6 対応には、2 つの意味がある。

1. IPv6 の Resource Record を持てること
2. IPv6 プロトコルで、クエリの送受信できること。

このどちらでも、RHEL5/CentOS の標準パッケージ(bind)で対応可能である。

IPv6 の **RR** 対応

IPv6 における Resource Record には、正引きに対する AAAA と逆引きに対する ip6.arpa がある。これは、bind8.4 もしくは bind9 移行で対応されている。

AAAA RR 設定：

```
;; example.jp
www     IN      A       192.0.2.1
        IN      AAAA    2001:db8::1
```

逆引き設定例：

named.conf

```
zone “0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa” {
        type master;
        file “2001.0db8.0000.0000.reverse”;
        allow-ransfer { slaves; };
        allow-query { any };
}
```

```
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0     IN      PTR     www.example.jp.
```

IPv6 プロトコルでのクエリの送受信

options linsten-on-v6 を指定 IPv6 プロトコルでのクエリの送受信が可能となる。

設定例：

```
options {
        directory “/var/named”;
        listen-on-v6 { any; };
…
```

ACL について

IPv4 と同様に、アドレスを直接記述することが可能である。

設定例：

```
acl “slaves” {
        192.0.2.1;       // slave server
        2001:db8::53;  // slave server
        127.0.0.1;       // for debug
        ::1;                 // for debug
};
```

リゾルバでの指定

IPv4 と同様に、/etc/resolv.conf、に直接 IPv6 アドレスを記述することで利用可能となる。

設定方法：

```
search examp.le.jp
nameserver 192.0.2.254
nameserver 2001:db8::53
```

確認方法

```
dig www.example.jp AAAA
dig -x 2001:db8::1
```

WEB について

Apache は、2.0 系から IPv6 に対応しているため、RHEL5/CentOS5 では、標準パッケージで IPv6 化が可能である。以降は、IPv6 に依存する箇所のみ記載する。

Listen ディレクティブ

アドレスは、IPv4 の時と異なり、ポート番号と区別するためにも、[] で括る必要がある。

設定例：
Listen [2001:db8::80]:80

ACL/アドレスによる制御

従来と同じく、IP アドレスをそのまま記述するが、特にネットワークアドレス指定時には、正確なネットワークアドレスを記述しなければならないことに注意が必要である。

設定例：
AuthName “Staff Only”
AutyType Basic
AuthUserFile “/var/www/www.example.jp/.htpasswd”
Require valid-user
Order Deny,Allow
Allow from 192.168.1.1
Allow from 2001:db8:0:1000::/64
Satisfy Any

上記の例では、2001:db8:0:1000::/64 からのアクセスを許可しているが、そのネットワークアドレスの指定に、2001:db8:0:1000/64 といった誤ったネットワークアドレスの指定では動作しないことに留意する必要がある。

アドレスベースの VirtualHost について

Listen ディレクティブと同様に、ポート番号と区別をつけるため、IPv6 アドレスは [] で括る必要がある。

設定例：
<VirtualHost [2001:db8:0:1000::80]:80>
  ServerName www.example.jp
…
..
</VirtualHost>

アクセスログについて

RHEL5/CentOS5 の Apache においては、IPv4 は、IPv4 射影アドレスとして取り扱われているが、アクセスログには、”::ffff:192.0.2.1” のようには出力されず、従来通り、IPv4 アドレスのまま出力される。

アクセスログ出力例：

2001:db8:0:1000:211:24ff:abcd:cafe -- [04/Mar/2011:09:30:59 +0900] “GET /favicon.ico HTTP/1.1” 404 272
192.0.2.1 - - [04/Mar/2011:09:30:59 +0900] “GET /favicon.ico HTTP/1.1” 404 272

SMTP

RHEL5/CentOS5 では、postfix により、IPv6 対応が可能となる。ここでは、IPv6 に依存した設定項目について記述する。

IPv6 への対応方法

/etcpostfix/main.cf

において、下記を記述し、postfix を再起動するだけで、対応完了となる。

inet_protocols = all

もしくは

inet_protocols = ipv4, ipv6

Listen するアドレスを明示的に指定

/etc/postfix/main.cf

inet_interfaces = 127.0.0.1. [::1], [2001:db8::25]

上記のように、inet_interfaces に “[]” 付きで、記述することで Listen アドレスを指定可能である。

送信時のアドレスを明示的に指定

複数の IPv6 アドレスを持っている場合など、送信時に明示的にどのアドレスを使うかを指定したい場合があるその際には

/etc/postfix/main.cf

において

smtp_bind_address6 = 2001:db8::25

と指定すればよい。この場合には、IPv6 アドレスの前後に “[]” は不要である。

[] で囲う必要があるもの

mynetworks や debug_peer_list のように、Postfix マッチリストが利用可能な場合、IPv6 アドレスに 含まれる “:” を利用している ”type:table” 形式と混乱しないためにも、IPv6 アドレスは [] で囲う必要がある。

例：
mynetworks = 127.0.0.0/8, [::1]/128, hash:/etc/postfix/network_table

POP3

RHEL5/CentOS5 では、標準パッケージである Dovecot によって、IPv6 対応が可能である。特に設定は不要であり、起動直後から、IPv6 が利用可能となっている。関連する設定項目は “Listen” であるが、標準パッケージのバージョンでは、Postfix のようなきめ細やかな設定はできない。

NTP

RHEL5/CentOS5 では、標準パッケージで IPv6 に対応することが可能となっている。

上位 NTP サーバの指定

従来と同じく、FQDN もしくは IPv6 アドレスによる指定が可能である。

設定例：
server ntp1.v6.exmaple.jp
server 2001:db8:0:1000::123

クエリの制限

reject コマンドでは、IPv4/IPv6 が扱えるが、IPv4 アドレスなら、”-4” を、IPv6 アドレスなら、”-6” をそれぞれ明示的に記述することが、推奨されている。その他の書式については、従来と同様である。

設定例：
restrict -4 192.0.2.1 mask 255.255.255.0 knod notrap nomodify nopeer noquery
restrict -6 2001:db8:0:: mask ffff:ffff:ffff:ffff:: knod notrap nomodify nopeer noquery

同期確認における注意点

従来通り、ntpq コマンドで実施可能だが、IPv4 アドレスしか表示することを考慮されていないので、最大 15 文字で、それ以上は切り捨てられてしまうことに注意する必要がある。つまり、どこと同期が撮れているかは、上記の prefix から類推するしかない。

同期前表示例

```
$ ntpq -pn
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 2001:3a0:0:2001 210.173.160.86   2 u   57   64    1    3.195    9.845   0.002
 2001:3a0:0:2005 210.173.160.56   2 u   56   64    1    3.173    9.871   0.002
```

同期後表示例

```
$ ntpq –pn
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 2001:3a0:0:2001 210.173.160.56   2 u   29   64   77    3.130    5.788   2.341
*2001:3a0:0:2005 210.173.160.86   2 u   33   64   77    3.173    9.871   4.693
```

SSH

サーバおよびクライアント共に RHEL5/CentOS5 の標準パッケージで IPv6 対応可能である。

SSH サーバについて

設定ファイルは、

/etc/ssh/sshd_config

である。通常の設定のままで、IPv6 に対応しているので、特に別途設定する必要はない。

SSH クライアントの利用について

FQDN を用いてアクセスする際には、特になにも意識せずに IPv4/IPv6 のどちらも利用できる。明示的に指定したい場合は

IPv4 で接続する場合
ssh -4 www.example.jp

IPv6 で接続する場合

ssh -6 www.example.jp

また IPv6 アドレスを直接記載することも可能である。

グローバルアドレスの場合
ssh 2001:db8:0:1000::80

リンクローカルアドレスの場合(スコープ ID を指定する必要 がある）
ssh fe80::abcd:ef%eth0

なおログイン後、どこから接続してきたのか last コマンドで調べる際、そのままだと IPv6 は、先頭部分しか表示されないが、”-a” オプションを指定することで、省略されることなく接続元の IPv6 アドレスが表示される。

syslog

RHEL5/CentOS5 に標準でインストールされている sysklogd は、IPv6 に対応していません。このため、IPv6 を利用したい場合は、rsyslog を導入する必要がある。

インストール例

CentOS5 でのインストール例
yum install rsyslog

リモートからログを受け取る

rsyslog は、初期状態ではネットワーク経由でログを受信することはできない。このため設定ファイルを修正する必要がある。

/etc/sysconfig/rsyslog

内において

SYSLOGD_OPTIONS=”-m 0”

という行があるので、それに”-r” + ポート番号を指定する。

設定例：
SYSLOGD_OPTIONS=”-r512 -m 0”

これで rsyslog を再起動させることで、IPv4/IPv6 のどちらでも受信できるようになる。なお、chkconfig 等で、syslog の自動起動を止め、rsyslog が自動起動するように修正するのを忘れないようにする必要がある。

アクセス制限について

IPv4 と同様に、AllowedSender に設定することで、送信元を制限することが可能 である。IPv6 アドレスは、[]で括る必要があり、なおかつ、ネットワークマスクは、[]内に含めてはならない。

設定例：
$AlloedSender UDP, 127.0.0.1, 192.0.2.0/24, [::1]/128, [2001:db8:0:1000::]/64

リモートに syslog を送信する方法について

リモートに syslog を飛ばすことも可能である。下記はすべてのログを 2001:db8:0:1000::512 に飛ばす例となる。IPv6 アドレスは[]で括る必要がある。それに続けて、ポート番号を指定することも可能である。

.　　@[2001:db8:0:1000::512]:512

これは、IPv4 において、TCP で送信する場合の設定方法でも同様である。