

Lightweight Languageの IPv6対応 PHP5編

運用設計ラボ合同会社
波田野 裕一

2013-11-26

1. PHPの概要

1.1. PHPの概要

■ PHPのバージョン系列

- 5.5系が最新系列になった。最新は5.5.6 (2013-11-14)
- 5.4系もメンテナンスされている。最新は 5.4.22 (2013-11-14)
- **5.3系は EoL (End of Life) になった。** (2013-07-11)
 - <http://php.net/archive/2013.php#id2013-07-11-1>

■ PHPとネットワークプログラミング

- 標準ライブラリで非常に広範囲なネットワークプログラミングが可能
その他の機能が欲しい場合は、拡張ライブラリ PEAR^{*1} のパッケージ等を使用する
 - PHP Extension and Application Repository <http://pear.php.net/>

■ PHPとIPv6

- PHP 5 から IPv6 に対応
 - <http://www.php.net/ChangeLog-5.php> (Version 5.0.0 Beta 1 のところ)

1.2. 各OSでのPHPのバージョン

OS・ディストリビューション	バージョン	PHP IPv6 対応状況	PHPバージョン
Linux	CentOS	5.10	○ distribution, updatesとも 5.3.3
		6.4	○ distribution, updatesとも 5.3.3
	Fedora	18	○ distribution, updatesとも 5.4.9
		19	○ distribution, updatesとも 5.5.0
	Debian	squeeze(6.x)	○ 5.3.3
		wheezy (7.x)	○ 5.4.4
	Ubuntu	12.04 LTS	○ distribution, updatesとも 5.3.10
		13.10	○ 5.5.3
FreeBSD	9.2	○ 5.4.21 (ports)	
Mac OS X	10.9	○ 5.4.17	

* 赤字はEoL(End of Life)のPHPパッケージ

2. PHPのIPv6対応

2.1. PHPのIPv6対応状況概略

考慮すべき要素	対応状況	備考
名前解決	○	<i>dns_get_record()</i> <i>gethostbyaddr()</i> <i>PEAR (Net_DNS2)</i>
ソケット	○	<i>inet_pton()</i> , <i>inet_ptop()</i> は <i>PHP5.1.0</i> 以降
各種 (L7) プロトコル	<i>HTTP</i> クライアント	○ 各種ファイル関数 <i>cURL</i> 、など
	<i>SMTP</i> クライアント	○ <i>PEAR (Net_Smtp)</i> システムの環境依存(<i>mail()</i> , <i>PEAR Mail</i>)
その他	IPv6アドレスの処理	○ <i>PEAR (Net_IPv6 拡張パッケージ)</i>

2.2. 名前解決 (1. 正引き)

■ dns_get_record()

- 引数で指定した RR の情報を取得して、配列で返す。

```
// internetweek.jp の IPv6 アドレス (AAAA レコード) を検索
$result = dns_get_record('internetweek.jp', DNS_AAAA);

foreach ( $result as $key_top => $array ) {
    echo "$key_top: ";
    foreach ( $array as $key => ${value} ) {
        echo " / $key: $value";
    }
    echo "<br>";
}
```

```
// $result は次のような配列になる
// array(
//   array(
//     "host" => "internetweek.jp",
//     "type" => "AAAA",
//     "ipv6" => "2001:dc2:1000:2006::c0:ffee",
//     "class" => "IN",
//     "ttl" => 300
//   )
// )
```

例えば、デュアルスタックのホストに対して、**DNS_ALL** を指定すると、AレコードとAAAAレコードの配列とそれぞれのhost / type / ip(ipv6) / class / ttlが連想配列(ハッシュ)で返ってくる。

逆引きも可能だが、arpa. 形式で指定する必要があるため、次頁の**gethostbyaddr()** を使う方が便利

2.2. 名前解決 (2. 逆引き)

■ `gethostbyaddr`

- ・ 引数で指定したIPアドレスに対応するホスト名を返す。(逆引き)
- ・ “::” で省略したアドレスを渡すことも可能

```
$result4 = gethostbyaddr('192.41.192.130');
```

```
$result6 = gethostbyaddr('2001:dc2:1000:2006::c0:ffee');
```

```
echo $result4; // internetweek.jp
```

```
echo $result6; // internetweek.jp
```


2.3. ソケット

■ PHPのソケット

- PHP5.0.0からは、基本的にIPv6に対応している。

■ PHP5.1.0以降に対応

- `inet_pton()`: IPアドレス変換 (テキスト形式 -> バイナリ形式)
- `inet_ntop()`: IPアドレス変換 (バイナリ形式 -> テキスト形式)

2.4. サービス (HTTP 1. ファイル関数-1)

■ fopen()

- php.ini で “allow_url_fopen = On” に設定している場合、fopen() でURLを指定して開くことができる。
 - サポートするプロトコル/ラッパー <http://jp1.php.net/manual/ja/wrappers.php>
- URL に IPv6 アドレスを使用する場合は **[2001:db8::1]** 形式で指定

<https://internetweek.jp/> のHTMLを取得し、表示するサンプル

```
$fh = fopen('https://[2001:dc2:1000:2006::c0:ffee]/', 'r');  
  
if ($fh === FALSE) { die("cannot open URL\n"); }  
  
while (($line = fgets($fh)) !== FALSE) {  
    echo $line;  
}  
  
fclose($fh);
```

2.4. サービス (HTTP 1. ファイル関数-2)

■ file() / file_get_contents()

- fopen()と同じく fopen_wrappers を使用している関数もIPv6に対応
 - **file()**: ファイル全体を配列として返す
 - **file_get_contents()**: ファイル全体を文字列として返す

https://internetweek.jp/ のHTMLを取得し、ソース1行毎にHRタグを挿入して出力するサンプル。

```
$contents = file("https://[2001:dc2:1000:2006::c0:ffee]/");  
if ($contents === FALSE) { die("cannot open URL\n"); }  
foreach ($contents as $line) { echo $line . <hr>; }
```

https://internetweek.jp/ のHTMLを取得し、表示するサンプル

```
$contents = file_get_contents('[2001:dc2:1000:2006::c0:ffee]');  
if ($contents === FALSE) { die("cannot open URL\n"); }  
echo $contents;
```

2.4. サービス (HTTP 2. HTTPクライアント)

■ cURL

- libcurl を使用し、各種プロトコルでのデータ転送を行なう。

URL の内容を文字列として取得する例

```
$ch = curl_init("https://internetweek.jp/");  
$fp = fopen("/tmp/internetweek.txt", "w");
```

```
curl_setopt($ch, CURLOPT_IPRESOLVE, CURL_IPRESOLVE_V6);  
curl_setopt($ch, CURLOPT_FILE, $fp);  
curl_setopt($ch, CURLOPT_HEADER, 0);
```

```
curl_exec($ch);  
curl_close($ch);  
fclose($fp);
```

2.4. サービス (SMTP 1. ビルトイン関数)

■ mail()

- システムのMTA(sendmail / postfix など)を使用してメールを送信
- 動作はMTAの動作環境に依存する。
 - ホストに IPv6 の接続性があるか
 - MTA が IPv6 を使用する設定になっているか

組み込み関数mail()でメール送信する例

```
mail($to, $subject, $message);
```

2.4. サービス (SMTP 2. PEAR-1)

■ PEAR::Mail

- 送信に使用するバックエンドの選択が可能
 - <http://pear.php.net/package/Mail>
- IPv6対応は利用するバックエンドに依存する。
 - PHP組み込み関数 mail()
 - sendmailプログラム
 - SMTPサーバに直接接続

バックエンドにSMTPサーバを利用する例

```
require 'Mail.php';  
  
$params = array('host' => $host);  
$m = Mail::factory('smtp', $params);  
$m->send($to, $header, $body);
```

2.4. サービス (SMTP 2. PEAR-2)

■ PEAR::Net_Smtp

- low-level の処理が可能な SMTP クライアント
 - http://pear.php.net/package/Net_Smtp
- OSのポリシーテーブルに従わないことに注意

SMTPサーバを\$hostに指定する例

```
require 'Net/SMTP.php';

$smarty = new Net_Smtp($host);
$smarty->connect();
$smarty->mailFrom($from);
$smarty->rcptTo($to);
$smarty->data($body);
$smarty->disconnect();
```

- デュアルスタックのホストに対しては IPv4 で接続
- IPv6 アドレスを指定する場合は [] で括る。

2.5. アドレス処理 (1. PEAR)

■ PEAR::Net_IPv6

- IPv6 アドレスに関する処理を行う。
 - http://pear.php.net/package/Net_IPv6
- 下記のような処理ができる
 - **checkIPv6()**: IPv6のアドレスとして正しいか検証
 - **compress()**: IPアドレスの短縮
 - **uncompress()**: IPアドレスの伸長

2.5. アドレス処理 (2. サンプル)

Net_IPv6::checkIPv6()

(::が2箇所ある誤った表記をチェックした例)

```
require 'Net/IPv6.php';
```

```
$result = Net_IPv6::checkIPv6('2001:db8:1::1');  
if ($result) { echo "TRUE"; }
```

```
$result = Net_IPv6::checkIPv6('2001:db8::1::1');  
if (! $result) { echo "FALSE"; }
```

Net_IPv6::compress() の例

```
require 'Net/IPv6.php';
```

```
echo Net_IPv6::compress('2001:db8:0:0:0:0:0:1');
```

```
echo Net_IPv6::compress('2001:db8:0:0:0123:0045:0006:07');
```

```
2001:db8::1
```

```
2001:db8::123:45:6:7
```

Net_IPv6::uncompress() の例

```
require 'Net/IPv6.php';
```

```
echo Net_IPv6::uncompress('2001:dc2:1000:2006::c0:ffee');
```

```
echo Net_IPv6::uncompress('2001:dc2:1000:2006::c0:ffee', TRUE);
```

```
2001:dc2:1000:2006:0:0:c0:ffee
```

```
2001:0dc2:1000:2006:0000:0000:00c0:ffee
```

第2引数を TRUE にすると、leading-zero が付与され、各コロンの間 が4文字の固定長になる

3. まとめ

PHP は 5 以降 IPv6 対応

ただ、5.3系以前のバージョンは EoL (End of Life)になったので、5.4系以降への移行が必要です。

参考文献

- Internet Week 2012 講演

「スクリプト言語とIPv6 -2012-」 / 関根 佳直 氏

<https://www.nic.ad.jp/ja/materials/iw/2012/proceedings/t7/t7-sekine.pdf>

- Internet Week 2011 講演

「Lightweight LanguageとIPv6」 / 関根 佳直 氏

<https://www.nic.ad.jp/ja/materials/iw/2011/proceedings/t5/t5-03.pdf>