diff cURL file_get_contents

目录

- cURL?

cURL是一种使用URL语法传输数据的开源命令行工具和库，支持FILE,FTP,HTTP(S)等多种协议。cURL为多种语言提供了API，其中就包括PHP。

- file_get_contents?

PHP内建函数，将一个文件实体读入字符串。也可抓取远程网页或API内容。（P.S. 对PHP函数命名也是醉了...）

	cURL	file_get_contents
易用性	【电钻】需扩展支持，语法复杂，有一系列函数实现相应功能。	【螺丝刀】PHP内建函数，语法简单。
功能性	功能齐全，通过setopt方法设置cURL会话选项，可以精细控制请求和响应的行为。出现错误时返回详细的错误信息，方便程序调试。	功能齐全，除了读取本地文件和抓取网页内容(GET)以外，通过stream context也可以实现HTTP POST,FILE,FTP和PHP等协议或方法。失败时返回false，不方便程序调试。
性能	抓取网页时，比FGC快30%左右。同时，对服务器的负载也更小。（P.S. 实践出出真知）	读取文件放在字符串中首选方法，使用内存映射技术（不占用真实内存，使用伪设备/dev/zero）增强性能。
并发	支持并发	不支持并发
杂项	当设置了open_basedir指令，cURL将不再支持file协议。	1. 读取本地文件时，如果使用相对路径且文件名外部可控，可导致任意文件读取漏洞。 2. 当allow_url_fopen=Off时，只可读取本地文件。

CURL的“坑”

- Expect: 100-continue

使用CURL发送>1024B的POST包时，请求会被分为2步：1，发送一个包含“Expect: 100-continue”头的请求；2，接收到100状态码后，发送POST包。有的服务器并不支持“Expect: 100-continue”，如：lighttpd。

```
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Expect:'));
```

- CURLOPT_NOSIGNAL

在(Li|U)nix中，当libcurl使用了标准的DNS解析时，设置<1000ms的超时时间，将会导致"CURL Error (28): Timeout was reached”错误。

```
curl_setopt($ch, CURLOPT_NOSIGNAL, true);
```

cURL并发实现

- rolling-curl: https://github.com/LionsAd/rolling-curl

- ParallelCurl: https://github.com/petewarden/ParallelCurl

走读ParallelCurl

```php
$options = array(
    CURLOPT_RETURNTRANSFER => true,     // return web page 返回网页
    CURLOPT_HEADER         => false,    // 不返回头信息
    CURLOPT_FOLLOWLOCATION => true,     // follow redirects
    CURLOPT_ENCODING       => "",       // handle all encodings
    CURLOPT_USERAGENT      => "spider", // 设置UserAgent
    CURLOPT_AUTOREFERER    => true,     // set referer on redirect
    CURLOPT_CONNECTTIMEOUT => 120,      // timeout on connect 连接超时
    CURLOPT_TIMEOUT        => 120,      // timeout on response 回复超时
    CURLOPT_MAXREDIRS      => 10,       // stop after 10 redirects
);
$ch = curl_init( $url );
curl_setopt_array( $ch, $options );
$content = curl_exec( $ch );
$err     = curl_errno( $ch );
$errmsg  = curl_error( $ch );
$header  = curl_getinfo( $ch );
if(isset($header['http_code']) && 500 == $header['http_code']){
    throw new Exception("cms-head-500");
}
curl_close( $ch );
```

```php
// WEBS-384 跳转到404页面
if (empty($this->routeId)) {
    $html404 = file_get_contents('http://www.tuniu.com/html/404.html');
    die($html404);
}
```

怎么选择？

- CURL

1，并发；

2，精细控制请求和响应的行为；

3，监控。

- file_get_contents

1，读取本地文件；

2，抓取网页或请求API；

3，简单发包，不关心返回。

参考页面

- http://curl.haxx.se/
- http://php.net/manual/en/book.curl.php
- http://php.net/manual/en/function.file-get-contents.php
- http://www.searchtb.com/2012/06/rolling-curl-best-practices.html
- http://www.laruence.com/2011/01/20/1840.html
- http://www.laruence.com/2014/01/21/2939.html
- https://mdb9.wordpress.com/2011/03/06/file_get_contents-vs-curl-what-has-better-performance/