

# [UNIX] PHP cURL Safe\_mode Bypass

---

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2008-01/msg00049.html>

---

- *From:* SecuriTeam <[support@xxxxxxxxxxxxxxxx](mailto:support@xxxxxxxxxxxxxxxx)>
  - *Date:* 24 Jan 2008 15:06:20 +0200
- 

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.

<http://www.securiteam.com/maillinglist.html>

-----

PHP cURL Safe\_mode Bypass

---

## SUMMARY

PHP is "an HTML-embedded scripting language. Much of its syntax is borrowed from C, Java and Perl with a couple of unique PHP-specific features thrown in. The goal of the language is to allow web developers to write dynamically generated pages quickly".

PHP supports libcurl, a library created by Daniel Stenberg, that allows you to connect and communicate to many different types of servers with many different types of protocols. libcurl currently supports the http, https, ftp, gopher, telnet, dict, file, and ldap protocols. libcurl also supports HTTPS certificates, HTTP POST, HTTP PUT, FTP uploading (this can also be done with PHP's ftp extension), HTTP form based upload, proxies, cookies, and user+password authentication.

## DETAILS

Vulnerable Systems:

- \* PHP version 5.2.4
- \* PHP version 5.2.5

Immune Systems:

## [UNIX] PHP cURL Safe\_mode Bypass

\* PHP version 5.2.6

The first issue [SAFE\_MODE bypass]

```
var_dump(curl_exec(curl_init("file://safe_mode_bypass\x00".__FILE__));
```

is caused by error in curl/interface.c

```
---
#define PHP_CURL_CHECK_OPEN_BASEDIR(str, len, __ret) \
if (((PG(open_basedir) && *PG(open_basedir)) || PG(safe_mode)) && \
strncasecmp(str, "file:", sizeof("file:") - 1) == 0) \
{ \
php_url *tmp_url; \
\
if (!(tmp_url = php_url_parse_ex(str, len))) { \
php_error_docref(NULL TSRMLS_CC, E_WARNING, "Invalid URL '%s'", str); \
php_curl_ret(__ret); \
} \
\
if (!php_memnstr(str, tmp_url->path, strlen(tmp_url->path), str + len)) { \
php_error_docref(NULL TSRMLS_CC, E_WARNING, "URL '%s' contains unencoded control characters", str); \
php_url_free(tmp_url); \
php_curl_ret(__ret); \
} \
\
if (tmp_url->query || tmp_url->fragment || \
php_check_open_basedir(tmp_url->path TSRMLS_CC) || \
(PG(safe_mode) && !php_checkuid(tmp_url->path, "rb+", CHECKUID_CHECK_MODE_PARAM)) \
) { \
php_url_free(tmp_url); \
php_curl_ret(__ret); \
} \
php_url_free(tmp_url); \
}
---
```

if you have `tmp_url = php_url_parse_ex(str, len)` where:  
`str = "file://safe_mode_bypass\x00".__FILE__`

and this function will return:

```
tmp_url->path = __FILE__
```

`curl_init()` functions checks safemode and openbasedir for `tmp_url->path`.  
Not for real path.

## [UNIX] PHP cURL Safe\_mode Bypass

```
if (argc > 0) {
char *urlcopy;

urlcopy = estrndup(Z_STRVAL_PP(url), Z_STRLEN_PP(url));
curl_easy_setopt(ch->cp, CURLOPT_URL, urlcopy);
zend_llist_add_element(&ch->to_free.str, &urlcopy);
}
---
```

the last step in curl\_init() function will only copy  
file://safe\_mode\_bypass to urlcopy.

The main problem exists in php\_url\_parse\_ex() function. If you put in  
curl\_init() "file://host/somewhere/path.php", php\_url\_parse\_ex() will  
select /somewhere/path.php to path variable. Looks good but it cannot be  
used, when you will check real path. Using file:///etc/passwd is correct  
but between file:// and /etc/passwd, php\_url\_parse\_ex() will select host  
and return path to /passwd.

```
cxib# php -v
PHP 5.2.5 with Suhosin-Patch 0.9.6.2 (cli) (built: Dec 10 2007 19:54:41)
(DEBUG)
Copyright (c) 1997-2007 The PHP Group
Zend Engine v2.2.0, Copyright (c) 1998-2007 Zend Technologies
```

How to fix  
CVS

<http://cvs.php.net/viewcvs.cgi/php-src/NEWS?revision=1.2027.2.547.2.1047&view=markup>  
<http://cvs.php.net/viewcvs.cgi/php-src/NEWS?revision=1.2027.2.547.2.1047&view=markup>  
Fixed a safe\_mode bypass in cURL identified by Maksymilian Arciemowicz.

### ADDITIONAL INFORMATION

The information has been provided by <mailto:cxib@xxxxxxxxxxxxxxxxxxxx>  
aksymilian Arciemowicz.

The original article can be found at:

[http://securityreason.com/achievement\\_securityalert/51](http://securityreason.com/achievement_securityalert/51)  
[http://securityreason.com/achievement\\_securityalert/51](http://securityreason.com/achievement_securityalert/51)

=====

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

list-unsubscribe@xxxxxxxxxxxxxxxx

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@xxxxxxxxxxxxxxxx

## [UNIX] PHP cURL Safe\_mode Bypass

=====  
=====

### DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.