Is there a way to execute php code in a sandbox from within php

Asked 16 years ago   Modified 9 years, 3 months ago   Viewed 15k times   ⬡ Part of PHP Collective

▲

24

▼

I want to execute a php-script from php that will use different constants and different versions of classes that are already defined.

Is there a sandbox php_module where i could just:

```
sandbox('script.php'); // run in a new php environment
```

instead of

```
include('script.php'); // run in the same environment
```

Or is [proc_open()](#) the only option?

PS: The script isn't accessible through the web, so fopen('http://host/script.php') is not an option.

`PHP` **php** **module** **sandbox**

Share Improve this question Follow

asked Nov 27, 2008 at 22:12

Bob Fanger
29.8k ● 7 ● 67 ● 82

6 Answers

Sorted by: | Highest score (default) ⬍ |

▲

10

▼

There is [runkit](#), but you may find it simpler to just call the script over the command line (Use [shell_exec](#)), if you don't need any interaction between the master and child processes.

Share Improve this answer Follow

answered Nov 27, 2008 at 22:23

troelskn
117k ● 27 ● 134 ● 156

About runkit; it does not seem to sandbox well by description, or I maybe should say easy. You can forbid functions, but I would rather like to forbid ALL but those in a provided list. If an user needs a function I can evaluate its safety by hand upon request. It seems like the only way is to write a custom made interpreter. If speed is an issue you can make it convert its AST into PHP or an other language, this is actually what I'm gonna do now as I could not find a ready solution. Cheers! Ps. Yes, I saw that this Q is old. Ds. – Frank Jun 14, 2010 at 15:38

Well .. Good luck finding something that can parse PHP into an AST ;) I concur with your point though. – troelskn Jun 14, 2010 at 17:27

FYI runkit seems to be abandoned and you'll probably want to compile either the CVS version or one of the patched versions (github.com/tricky/runkit) if you want to run it on a modern version of PHP – Eli Jul 21, 2010 at 14:21

▲

6

▼

🔖

↺

The is a class on GitHub that may help, early stages but looks promising.

https://github.com/fregster/PHPSandbox

Share Improve this answer Follow

answered Jul 2, 2011 at 10:35

Paul
71 ● 1 ● 1

▲

2

▼

🔖

↺

Also, you should look at the backtick operator:

```
$sOutput = `php script_to_run.php`;
```

This will allow you to inspect the output from the script you are running. However, note that the script will be run with the privileges you have, but you can circumvent this by using sudo on Linux.

This approach also assumes that you have the PHP CLI installed, which is not always the case.

Share Improve this answer Follow

answered Nov 28, 2008 at 11:04

Vegard Larsen
13k ● 14 ● 60 ● 102

There is `Runkit Sandbox` - you might get it to work, it's a PHP extension. I'd say the way to go.

But you might need to create a "sandbox" your own, e.g. by resetting the global variable state of the superglobals you use.

```php
class SandboxState
{
    private $members = array('_GET', '_POST');
    private $store = array();
    public function save() {
        foreach($members as $name) {
            $this->store[$name] = $$name;
            $$name = NULL;
        }
    }
    public function restore() {
        foreach($members as $name) {
            $$name = $this->store[$name];
            $this->store[$name] = NULL;
        }

    }
}
```

Usage:

```php
$state = new SanddboxState();
$state->save();

// compile your get/post request by setting the superglobals
$_POST['submit'] = 'submit';
...

// execute your script:
$exec = function() {
    include(func_get_arg(0)));
};
$exec('script.php');

// check the outcome.
...

// restore your own global state:
$state->restore();
```

Share Improve this answer Follow

answered Apr 17, 2012 at 10:03

hakre
197k ●55 ●446 ●853

1

I have developed a BSD-licensed sandbox class for this very purpose. It utilizes the PHPParser library to analyze the sandboxed code, check it against user-configurable whitelists and blacklists, and features a wide array of configuration options along with sane default settings. For your needs you can easily redefine classes called in your sandboxed code and route them to different ones.

The project also includes a sandbox toolkit (use only on your local machine!) that can be used to experiment with the sandbox settings, and a full manual and API documentation.

https://github.com/fieryprophet/php-sandbox

Share Improve this answer Follow

answered Mar 4, 2013 at 5:18

Elijah Horton
21 ● 2

> Is there anyway to use your class without installing it via composer. I dont want to install it on the live server. Just using the plain classes? – Obaid Maroof Oct 7, 2013 at 10:11

0

i know its not 100% topic related, but maybe useful for somebody n__n

```
function require_sandbox($__file,$__params=null,$__output=true) {

    /* original from http://stackoverflow.com/a/3850454/209797 */

    if($__params and is_array($__params))
     extract($__params);

    ob_start();
    $__returned=require $__file;
    $__contents=ob_get_contents();
    ob_end_clean();

    if($__output)
     echo $__contents;
    else
     return $__returned;

};
```

Share

Improve this answer

Follow

edited Apr 11, 2014 at 6:53

answered Jan 12, 2013 at 4:33

AgelessEssence
6,733 ● 5 ● 35 ● 38

> it doesn't work if $__file tries to define already defined constants. it may also breaks global variables if $__file uses the global keyword or the $GLOBALS array to reference variables in the global scope. – DrLightman Apr 11, 2014 at 0:09

i updated the code, try it now :D, it is useful for me :D may be not perfect, but is useful for me to require some php files who create vars during their execution that i dont want be in global scope :D – AgelessEssence Apr 11, 2014 at 6:50 ✎

This is more of a template_require, not really a sandbox – Tofandel Apr 7, 2021 at 9:52