The Basics: PHP register_globals Overview (http://dan.doezema.com/2010/04/php-register-globals-overview)

APRIL 27, 2010

In this article I'll explain what register_globals is; how to protect against exploits that take advantage of it; and why it should be turned off (if possible).

What is Register Globals?

`register_globals` is a setting/feature within PHP that was intended to ease development by making variables passed to the script (via a form, cookie, or session) automatically available as predefined variables within the global scope.

In the example below you can see how `register_globals` takes a variable from the page's query string and creates the global variable `$apple` to represent it.

Page: `index.php?apple=red`

```
echo $apple; // 'red'
```

How Register Globals Can Be Exploited

At first glance one might say *“That’s a great feature! Now I don’t have to go to the trouble of defining* `$apple` *and assigning it a value!”*

In my time as a web developer I’ve come across a few great pieces of advice.

- Never trust the user.
- Don’t assume.
- If it *can* happen, it *will* happen.

Below is a excerpt from a Shopping Cart script; watch what happens when we *assume*…

Page: `cart.php?promo_code=12save`

```
if($promo_code == '12save') {
  $discount= 0.10;
}

if(isset($discount)) {
  $price -= $price * $discount;
}
```

This might look secure to a new or even intermediate PHP developer…

I mean, come on… the only way to get the discount is by knowing the correct promo code, right?!

Wrong Let’s add another variable to the page’s query string and see what happens…

Page: `cart.php?promo_code=doesNotMatter&discount=0.80`

```
if($promo_code == '12save') {
  $discount= 0.10;
}

/**
 * Even though the promo code was incorrect this
 * IF statement will still evaluate TRUE and discount
 * the price.
 */
if(isset($discount)) {
  $price -= $price * $discount;
}
```

The reason this exploit works is because `register_globals` has defined `$promo_code` and `$discount` based on the page's query string **before** any of the script's code was executed.

So how can one combat this? Easy, *don't assume* what the value of `$discount` will be, explicitly set it to `0` by default.

Page: `cart.php?promo_code=12save`

```
$discount = 0;
if($promo_code == '12save') {
  $discount = 0.10;
}

if(isset($discount) && ($discount > 0)) {
  $price -= $price * $discount;
}
```

Be aware that `register_globals` will allow keys => values to be inserted into existing array variables – this is often missed by developers of all skill levels. Array keys used later in a script must be defined with a default value to avoid exploitable code.

Below is an example of an attacker successfully forcing a "debug" mode, thus allowing him/her to see PHP errors.

Page: `index.php?config[debug]=1`

```
if(isset($config['debug']) && ($config['debug'] == true)) {
  error_reporting(E_ALL);
}
```

Why Register Globals Should Be Turned Off

Having `register_globals` enabled is like playing with fire. It’s a crutch used by new or intermediate PHP developers that don’t know any better – or in some cases are just too lazy to use proper coding practices.

Since PHP 4.2.0 `register_globals` has been disabled by default; and in PHP 5.3 it’s now considered deprecated. However, many shard hosting companies keep it enabled on their servers as they host older sites that were developed during a time of heavy `register_globals` reliance.

If you are starting a new project on a server where you have access to the `php.ini` file I would suggest you turn off `register_globals`.

If you don’t have access to the `php.ini` file; you’re on a shard server; or you have other sites on your server that break when you turn `register_globals` off you can try adding the following line of code to the root directory’s `.htaccess` file…

File: `.htaccess`

```
php_flag register_globals off
```

Summary

It’s important to realize that `register_globals`, by itself, is **not** a security flaw within PHP. If a developer is following proper coding practices there is no need to worry about `register_globals` being on or off.

Remember to always define your variables and array keys before you use them!

SECURITY (HTTP://DAN.DOEZEMA.COM/TAGS/#SECURITY) PHP (HTTP://DAN.DOEZEMA.COM/TAGS/#PHP)

REGISTER_GLOBALS (HTTP://DAN.DOEZEMA.COM/TAGS/#REGISTER_GLOBALS)

THE BASICS: PHP REGISTER_GLOBALS OVERVIEW (HTTP://DAN.DOEZEMA.COM/2010/04/PHP-REGISTER-GLOBALS-OVERVIEW) WAS PUBLISHED ON APRIL 27, 2010