

Forum:[Hack Forums](#)

Thread: [How to exploit with \[WordPress Core 5.8.2 - 'WP_Query' SQL Injection\]](#)

night4pe - July 20, 2023, 4:45 am

I have readed post Following link.

[WordPress Core 5.8.2 - 'WP_Query' SQL Injection](#)

But in the below statements i cant understand that.

Code

```
action=[u]<action_name>[/u]&nonce=a85a0c3bfa&query_vars={"tax_query":{"0":{"field":"term_taxonomy_id","terms":["[u]<inject>[/u]"]}}}
```

I need your help

Money- - July 20, 2023, 5:02 am

(Jul 20, 2023 04:45 AM)night4pe Wrote: [\[CITE\]](#) I have readed post Following link.

[WordPress Core 5.8.2 - 'WP_Query' SQL Injection](#)

But in the below statements i cant understand that.

Code

```
action=[u]<action_name>[/u]&nonce=a85a0c3bfa&query_vars={"tax_query":{"0":{"field":"term_taxonomy_id","terms":["[u]<inject>[/u]"]}}}
```

I need your help

You need to send a POST Request to this page

<http://target.com/wp-admin/admin-ajax.php>

with the following body

```
action=<action_name>&nonce=a85a0c3bfa&query_vars={"tax_query":{"0":{"field":"term_taxonomy_id","terms":["<inject>"]}}}
```

The inject part you can do blind , union or time based SQLI , i think time-based would be best on this API or JSON body since i dnt think anything will display back as a response or maybe I am wrong.

Try manual SQLI or try SQLMAP to see if it works.

InspectorGadget - July 20, 2023, 5:06 am

(Jul 20, 2023 05:02 AM)Money- Wrote: [\[CITE\]](#) You need to send a POST Request to this page

<http://target.com/wp-admin/admin-ajax.php>

with the following body

action=<action_name>&nonce=a85a0c3bfa&query_vars={"tax_query":{"0":{"field":"term_taxonomy_id"},"terms":["<inject>"]}}}

The inject part you can do blind , union or time based SQLI , i think time-based would be best on this API or JSON body since i dnt think anything will display back as a response or maybe I am wrong.

Try manual SQLI or try SQLMAP to see if it works.

This advice is solid, but 'd also take not of the <action_name> part, you'll need to replace that with a valid action name otherwise it will still fail.

night4pe - July 20, 2023, 5:56 am

(Jul 20, 2023 05:06 AM)InspectorGadget Wrote: [\[CITE\]](#) This advice is solid, but 'd also take not of the <action_name> part, you'll need to replace that with a valid action name otherwise it will still fail.

what are the correct values for parameter action_name and inject?

Money - July 20, 2023, 6:33 am

(Jul 20, 2023 05:56 AM)night4pe Wrote: [\[CITE\]](#) what are the correct values for parameter action_name and inject?

You can google or check the web for the values. I think this is something from the admin panel , try download the source code of that wordpress version you are targeting and check if there's anything u can put on the actions.

Its some sort of admin dashboard function or something. Download the open source wordpress files btw not download them from the target.

Or just google that CVE on github or other places to find a better explanation with solid examples of the POC.

night4pe - July 20, 2023, 7:30 am

Helped a lot.

Thank you.

m3m0o - June 5, 2024, 6:41 am

(Jul 20, 2023 07:30 AM)night4pe Wrote: [\[CITE\]](#) Helped a lot.

Thank you.

Dude, did u managed to exploit this vulnerability? I'm in a Wordpress 5.8.2 site. Seems vulnerable but I can't find a valid unauthenticated action that uses WP_Query. The only two actions available are heartbeat and generate_password. I don't really know if I need to find some other unauthenticated action the plugins to work out. Can u help me?

G.P.D. - June 6, 2024, 3:35 pm

Haven't tried it but here is a PoC for that CVE :

<https://github.com/sealldeveloper/CVE-2022-21661-PoC>

By looking into the source code I can tell that it uses the action "test".

m3m0o - June 7, 2024, 3:59 am

(Jun 6, 2024 03:35 PM)G.P.D. Wrote: [\[CITE\]](#) Haven't tried it but here is a PoC for that CVE :

By looking into the source code I can tell that it uses the action "test".

Yes, I have tried all these available POCs but none of them work. All actions that are present do not work and the endpoint returns error 400. The only two builtin actions available to unauthenticated users are heartbeat and generate_password, but they have no relation to the database. I believe

that I will have to create a test environment with Wordpress in this version and all the plugins that are on the site in question, that way I will have access to the source code and can try to look for a plugin or theme that has a valid action. I don't know how this PoC, with the "test" action, works. It is not a valid action. As far as I read, there was no workaround for this vulnerability other than updating the Wordpress version, so I can't understand how that works in the video PoC but not on my target website.