1

Followup - SQL Injection - https://[redacted]/[redacted]/MSI.portal

Share:

State	○ Resolved (Closed)
Disclosed	**May 14, 2020 10:37pm +0530**
Reported To	U.S. Dept Of Defense
Weakness	SQL Injection
Severity	High (7 ~ 8.9)
Participants	
Visibility	Disclosed (Full)

Collapse

TIMELINE · EXPORT

arkaic submitted a report to **U.S. Dept Of Defense**. Sep 11th (9 months ago)

Summary:

Time based blind sql injection for parameter MSI_additionalFilterType1, at the following URL:

https://[redacted]/[redacted]/MSI.portal?_nfpb=true&_pageLabel=msi_portal_page_61

Description:

This is a follow up to a previous report I submitted:

https://hackerone.com/reports/674838

The following page has a form parameter which is vulnerable to time based blind sql injection, which allows an attacker to retrieve information from the database.

https://[redacted]/[redacted]/MSI.portal?_nfpb=true&_pageLabel=msi_portal_page_61

The page uses several hidden parameters which are sent when the form is submitted. The specific vulnerable parameter in this case is "MSI_additionalFilterType1".

Sample form POST data, prior to SQL injection testing:

https://[redacted]/[redacted]/msi/query_results.jsp?MSI_additionalFilterType1=-999&MSI_additionalFilterType2=-999&MSI_additionalFilterValue1=-999&MSI_additionalFilterValue2=-999&MSI_generalFilterType=-999&MSI_generalFilterValue=-999&MSI_outputOptionType1=-999&MSI_outputOptionType2=-999&MSI_outputOptionValue1=-999&MSI_outputOptionValue2=-999&MSI_queryType=-999

Initially I was not able to retrieve details about the database user nor the schema. After adjusting several parameters for sqlmap, I was able to successfully do so.

Here we can see the specific edition of Oracle DB used, along with the user and database name:

[redacted]

```
banner: 'Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production'
[13:11:58] [INFO] fetching current user
[13:11:58] [INFO] retrieved: [redacted]
current user: '[redacted]'
[13:13:17] [INFO] testing if current user is DBA
current user is DBA: True
[13:13:25] [WARNING] schema names are going to be used on Oracle for enumeration as the counterpart to dat
[13:13:25] [INFO] fetching database (schema) names
[13:13:25] [INFO] fetching number of databases
[13:13:25] [INFO] retrieved:
[13:13:29] [WARNING] in case of continuous data retrieval problems you are advised to try a switch '--no-c
[13:13:29] [ERROR] unable to retrieve the number of databases
[13:13:29] [INFO] falling back to current database
[13:13:29] [INFO] fetching current database
[13:13:29] [INFO] retrieved: [redacted]
```

```
[13:14:48] [WARNING] on Oracle you'll need to use schema names for enumeration as the counterpart to datab
available databases [1]:
[*] [redacted]
```

Here you can see the retrieval of a few table names from the database:

[redacted]

```
[13:18:06] [INFO] fetching tables for database: '[redacted]'
[13:18:06] [INFO] fetching number of tables for database '[redacted]'
multi-threading is considered unsafe in time-based data retrieval. Are you sure of your choice (breaking w
[13:18:08] [INFO] retrieved:
[13:18:14] [INFO] adjusting time delay to 3 seconds due to good response times
67
[13:18:32] [INFO] retrieved: [redacted]
[13:19:54] [INFO] retrieved: [redacted]
[13:23:29] [INFO] retrieved: [redacted]
[13:25:45] [INFO] retrieved: [redacted]
[13:28:37] [INFO] retrieved: [redacted]
```

I interrupted the process at this point, so as to not enumerate all 67 table names, and ceased further testing.

Impact

High

Step-by-step Reproduction Instructions

1. Visit the vulnerable url (https://[redacted]/[redacted]/MSI.portal?_nfpb=true&_pageLabel=msi_portal_page_61) while using an intercepting proxy
2. Intercept GET request to capture full URL and all form parameters
3. Utilize sqlmap to detect and exploit sql injection in "MSI_additionalFilterType1" parameter

Note: The default configuration of sqlmap will not be able to find the sql injection. I adjusted the following parameters in order to do so. "--risk 2 --level 3" and "--tamper=space2comment,randomcase,between"

Product, Version, and Configuration (If applicable)

Suggested Mitigation/Remediation Actions

1. Sanitize all form parameter inputs, and use whitelisting to allow only needed data
2. Rate limit submissions of forms. Time based sql injection requires many more HTTP requests than would be seen from legitimate browser activity.

Impact

High/Critical impact.

This sql injection attack could be used to retrieve all information from the database. Also, the account is running with DBA privileges which would allow for the retrieval of database account passwords and takeover of the server itself via injection of system commands; these could be leveraged to attack other systems on the network and potential lateral movement to other systems.

BOT: U.S. Dept Of Defense posted a comment. Sep 11th (9 months ago)

Greetings from the Department of Defense (DoD),

Thank you for supporting the DoD Vulnerability Disclosure Program (VDP).

By submitting this report, you acknowledge understanding of, and agreement to, the DoD Vulnerability Disclosure Policy as detailed at @DeptofDefense.

The VDP Team will review your report to ensure compliance with the DoD Vulnerability Disclosure Policy. If your report is determined to be out-of-scope, it will be closed without action.

We will attempt to validate in-scope vulnerability reports and may request additional information from you if necessary. We will forward reports with validated vulnerabilities to DoD system owners for their action.

Our goal is to provide you with status updates not less than every two weeks until the reported vulnerability is resolved.

Regards,

The VDP Team

arkaic posted a comment. Sep 11th (9 months ago)

Oops. The screenshot are in the wrong order for the text they are meant to illustrate.

deez_nops updated the severity to High. Sep 11th (9 months ago)

deez_nops changed the status to ○ **Triaged**. Sep 11th (9 months ago)

Greetings,

We have validated the vulnerability you reported and are preparing to forward this report to the affected DoD system owner for resolution.

Thank you for bringing this vulnerability to our attention!

We will endeavor to answer any questions the system owners may have regarding this report; however, there is a possibility we will need to contact you if they require more information to resolve the vulnerability.

You will receive another status update after we have confirmed your report has been resolved by the system owner. If you have any questions, please let me know.

Thanks again for supporting the DoD Vulnerability Disclosure Program.

Regards,

The VDP Team

agent-l8 changed the status to ○ **Needs more info**. Dec 5th (6 months ago)

Greetings,

Can you please answer the following questions?

I am unable to verify if this has been fixed as there is not a full PoC command on what you are using for sqlmap. I have tried a few ways, but can you confirm if this is fixed? That way we can close or push back to tell them to fix.

If you have any questions, please let me know.

Thanks again for supporting the DoD Vulnerability Disclosure Program.

Regards,

arkaic changed the status to ○ **New**. Dec 5th (6 months ago)

@agent-l8 Hello. It appears that the site has been redesigned, and that it is now under Akamai WAF protections. I'd like to take a couple of days to explore the changes, in order to determine if there are any bypasses.

arkaic posted a comment. Dec 7th (6 months ago)

@agent-l8 I was able to spend some time on this, and the issue appears to be fixed. The vulnerable form POST is no longer present, and Akamai is effectively blocking other attempts at SQL injection. I feel that this can be closed. If closed, I would appreciate disclosure, redacted where necessary of course, of my finding.

agent-l8 posted a comment. Dec 10th (6 months ago)

Hey @arkaic , sounds good- that's why I was unable to reproduce. We are unable to close the report without a remediation notification from the system owner. Once that is done, you may request disclosure (though be aware it is taking us awhile right now as we are dealing with a process change). Thanks!

agent-l8 closed the report and changed the status to ○ **Resolved**. Dec 20th (5 months ago)

Good news!

The vulnerability you reported has been resolved and this report is now closed. If you have any further questions or disagree that the report is resolved, please let us know.

Thank you for your time and effort to improve the security of the DoD information network.

Regards,

The VDP Team

arkaic posted a comment. Dec 20th (5 months ago)

Great! I'd still like to request disclosure for this issue.

agent-r5 posted a comment. Dec 20th (5 months ago)

We can add it to the queue. Just toggle the request for disclosure from your side so we can track it.

arkaic requested to disclose this report. Dec 20th (5 months ago)

ag3nt-j1 agreed to disclose this report. May 14th (17 days ago)

Thanks for participating in the DoD Vulnerability Disclosure Program. We're publicly disclosing your resolved report at this time. We also lock the report on the disclosure but feel free to reach out to us at VDP-Questions@dc3.mil if there are any questions, concerns or issues.

This report has been disclosed. May 14th (17 days ago)

U.S. Dept Of Defense has locked this report. May 14th (17 days ago)