ISSUE Found

SQL Injection

CWE ID

89

Affected URL: -

1. <https://apisetu.gov.in/public/console/marketplace/directory%23/?%23Ration%2520Card=&Mostpopular=RATCR%27+AND+%271%27%3D%271%27+-->+

Method GET

Parameter Mostpopular

Attack RATCR' OR '1'='1' --

1. <https://apisetu.gov.in/public/console/marketplace/directory%23/?%23Driving%2520License=&Mostpopular=DRVLC+AND+1%3D1+-->+

Method GET

Parameter Mostpopular

Attack DRVLC OR 1=1 –

How to Fix: -

1. Do not trust client side input, even if there is client side validation in place.
2. In general, type checks all data on the server side.
3. If the application uses JDBC, use PreparedStatement or CallableStatement, with parameters passed by '?'
4. If the application uses ASP, use ADO Command Objects with strong type checking and parameterized queries.
5. If database Stored Procedures can be used, use them.
6. Do \*not\* concatenate strings into queries in the stored procedure, or use 'exec', 'exec immediate', or equivalent functionality!
7. Do not create dynamic SQL queries using simple string concatenation.
8. Escape all data received from the client.
9. Apply an 'allow list' of allowed characters, or a 'deny list' of disallowed characters in user input.
10. Apply the principle of least privilege by using the least privileged database user possible.
11. In particular, avoid using the 'sa' or 'db-owner' database users. This does not eliminate SQL injection, but minimizes its impact.
12. Grant the minimum database access that is necessary for the application.

More information: -

1. The page results were successfully manipulated using the boolean conditions [RATCR' AND '1'='1' -- ] and [RATCR' OR '1'='1' -- ]
2. The parameter value being modified was NOT stripped from the HTML output for the purposes of the comparison
3. Data was NOT returned for the original parameter.
4. The vulnerability was detected by successfully retrieving more data than originally returned, by manipulating the parameter

References: -

1. <https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html>