**TempTalk BUG REPORT**

|  |  |
| --- | --- |
| Report Title | SQL Injection on Login Page |
| Reporter | John Doe (john.doe@gmail.com) |
| Submit Date | 30 February 2024 |
| Affected Endpoint | https://chative.com |
| Severity | Medium (CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:L/VI:L/VA:L/SC:L/SI:L/SA:L) |
| Proof of Concepts | [PoC Video Link](https://youtu.be/dQw4w9WgXcQ?si=l8r4a6VXy0l7p37T) |

**Summary**

This is a SAMPLE bug report. You can use this template to submit your findings to our bug bounty program.

I found a SQL injection vulnerability in the website's Login page. I can use this vulnerability to obtain the email and password of the website's registered users, and use these emails and passwords to log in to any user's account. I can also use this vulnerability to obtain server permissions and attack more servers.

**Steps to Reproduce**

1. Explain briefly how to reproduce the finding.Please describe the penetration testing process as detailed as possible so that the vulnerability can be quickly reproduced during the audit and the results can be fed back to you.
2. If a test account is used during testing and the account is special, you need to provide the test account in the report to facilitate the auditor to reproduce the vulnerability.
3. You need to provide key screenshots during the penetration test, such as obtaining server permissions, successfully connecting to the database, etc.
4. If there are weak passwords or passwords obtained through brute force in the system, you need to indicate this in the report.
5. Please indicate the link to the POC/EXP or other tools you used in the penetration test. If you modified part of the POC/EXP, you need to indicate the modified content in the report.

**Expected Result**

Explain the expected proper result / output that should be done if the finding is fixed.

For example, when entering a SQL statement into the query, all registered user information should not be displayed.

**Actual Result**

Explain the actual result / output from this finding.

For example, when logging in with an SQL statement, you can obtain information about all registered users by constructing a specific statement.

**Impact**

Explain briefly about the impact of this finding, and some recommendation on how to fix it.

Attackers can use SQL injection vulnerabilities to view database information, obtain a large amount of user information, and even obtain server permissions.

Defense against SQL injection vulnerabilities can be achieved by filtering keywords or escaping all characters that have a special meaning in SQL.

**Notes**

Any additional information that you like to add.