Muhammed Raihan P A

17 July 2021

Reported to picoctf.org

# Credentials are checked at the client side

**SUMMARY :-**

‘**Irish-Name-Repo 2**’ challenge in picoCTF challenges was vulnerable to SQLinjection attacks. I could bypass the login page by using some SQLi payloads and by commenting the password field out.

**STEPS :-**

1. I tried to bypass using many payloads like **‘ OR 1 -- -," OR 1 = 1 -- -,OR 1=1** etc. None worked.
2. Most probably the common payloads have been prevented from SQLi attacks,and the password field has been filtered.
3. So I tried commenting out the password field. I used the payload **admin’--**
4. Yeah! It worked. I got the flag as a message **picoCTF{m0R3_SQL_plz_c34df170}**

**IMPACT :**

By using SQLinjection anyone can login as admin and can take over the whole database.

**MITIGATION :**

- The most important precautions are data sanitization and validation, which should already be in place. Sanitization usually involves running any submitted data through a function (such as MySQL's mysql_real_escape_string() function) to ensure that any dangerous characters (like " ' ") are not passed to a SQL query in data.
- Validation is slightly different, in that it attempts to ensure that the data submitted is in the form that is expected. At the most basic level this includes ensuring that e-mail addresses contain an "@" sign, that only digits are supplied when integer data is expected, and that the length of a piece of data submitted is not longer than the maximum expected length
- Don't use dynamic SQL when it can be avoided. use prepared statements, parameterized queries or stored procedures instead whenever possible.
- Blacklist maximum known payloads