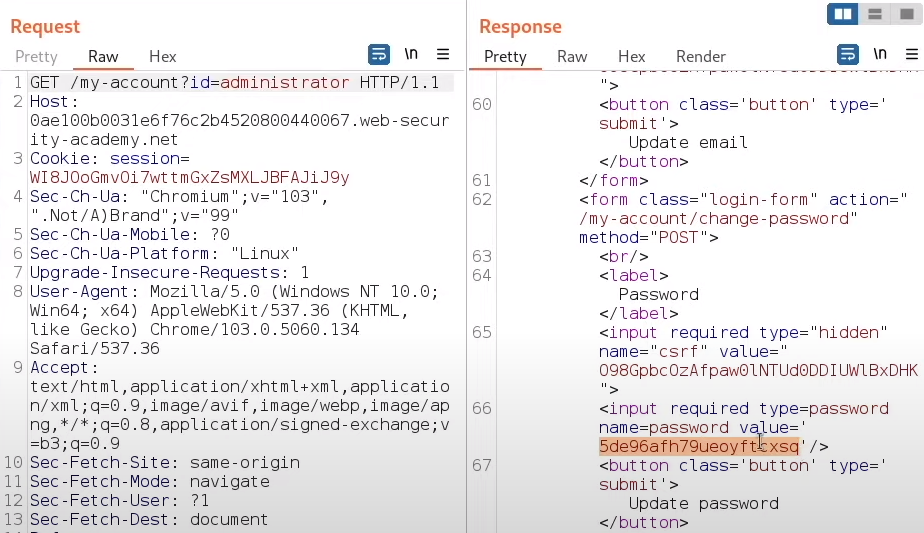
**SCENARIO**

This lab has a user account page that displays the current user's existing password, prefilled in a masked input. The goal is to retrieve the administrator's password and then use it to delete the user 'carlos'. For testing purposes, you have access to your own account with the credentials: wiener:peter.

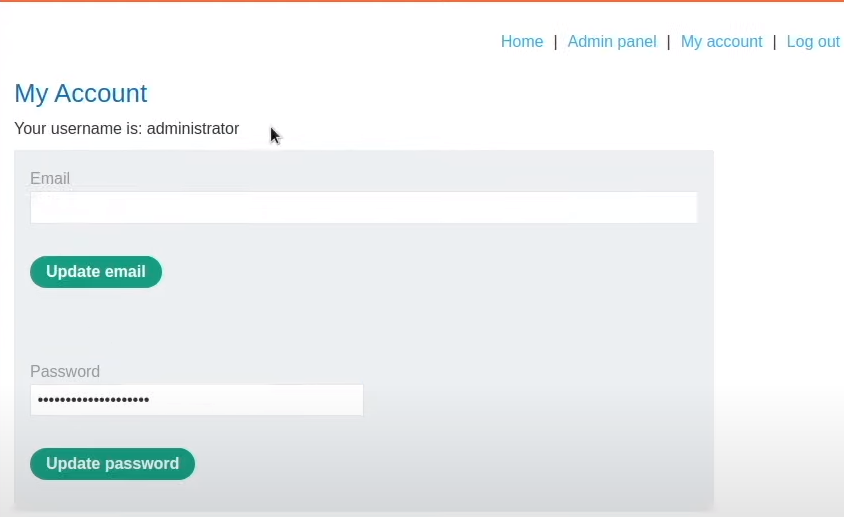
**PROCEDURE**

1. Log in to the application using the provided credentials: wiener:peter.
2. Once logged in, navigate to the user account page.
3. In the address bar, modify the "id" parameter to target the 'administrator' instead of your own user ID.
4. Intercept the response using Burp Suite or a similar proxy tool.
5. Observe the response, specifically noting the leaked administrator's password.
6. Using the retrieved administrator password, log in as the administrator.
7. Navigate to the user management or relevant section and delete the user 'carlos'.

**PAYLOAD**

The primary payload here involves changing the "id" parameter to target another user, in this case: id=carlos.

**PROOF OF CONCEPT**



**REMEDIATION**

1. **Avoid Displaying Sensitive Information:** Passwords, even if masked, should not be displayed in user interfaces.
2. **Implement Proper Authorization Checks:** Ensure that users can only access and modify their own account details and not those of others by merely changing parameters.
3. **Encrypt or Hash Passwords:** If for any reason a password must be displayed or transmitted, it should be encrypted or hashed, and the original should never be retrievable.
4. **Implement Parameterized Queries:** This helps in preventing unauthorized data access or manipulation.