**SCENARIO**

This lab is vulnerable to password reset poisoning. The user 'carlos' is known to indiscriminately click on any links in emails he receives. The objective is to exploit this vulnerability and log in to Carlos's account. Users can access their own account with the credentials: wiener:peter. Emails sent to this account can be viewed via the email client on the exploit server.

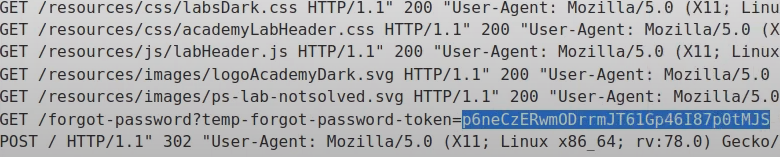
**PROCEDURE**

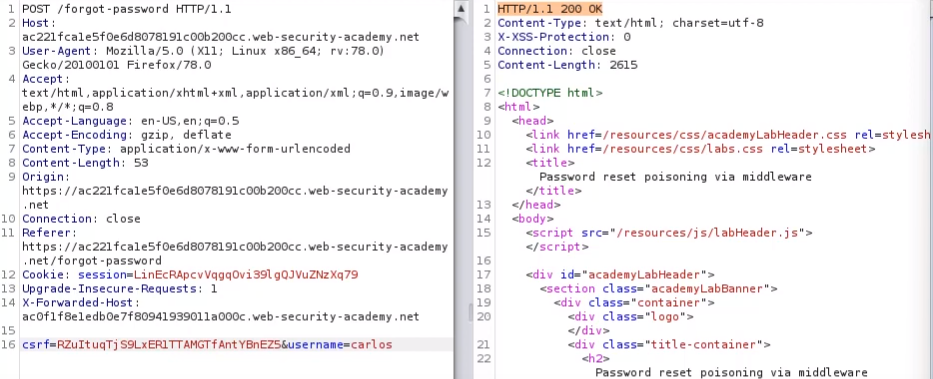
1. While having Burp Suite active, explore the password reset function and observe the unique reset token link sent via email.
2. Intercept the POST /forgot-password request and forward it to Burp Repeater. Note the support for the X-Forwarded-Host header, which allows redirecting the reset link to an arbitrary domain.
3. Visit the exploit server and note down the exploit server URL.
4. Return to the Burp Repeater request and add the X-Forwarded-Host header, using the exploit server URL:

makefileCopy code

1. X-Forwarded-Host: YOUR-EXPLOIT-SERVER-ID.exploit-server.net
2. Modify the 'username' parameter to 'carlos' and transmit the request.
3. Navigate to the exploit server and inspect the access log. Here, you should identify a GET /forgot-password request with the victim's token as a query parameter. Record this token.
4. Return to your email client, copy the legitimate password reset link (excluding the one redirected to the exploit server), paste it into a browser, and adjust the value for the 'temp-forgot-password-token' parameter to the token you acquired from the victim.
5. Access this URL to set a new password for Carlos's account.
6. Finally, use the new password to log into Carlos's account, thus solving the lab.

**PAYLOAD**

**PROOF OF CONCEPT  
**

****

**REMEDIATION**

1. **Secure Headers:** Don't trust or act upon the X-Forwarded-Host header or any other headers that can be controlled by the user.
2. **Token Validation:** Implement strict validation for password reset tokens ensuring they cannot be easily guessed or hijacked.
3. **Limit Reset Attempts:** Implement rate limiting on password reset requests to prevent abuse.
4. **Security Awareness Training:** Educate users like 'carlos' on the dangers of mindlessly clicking on links in unsolicited emails.