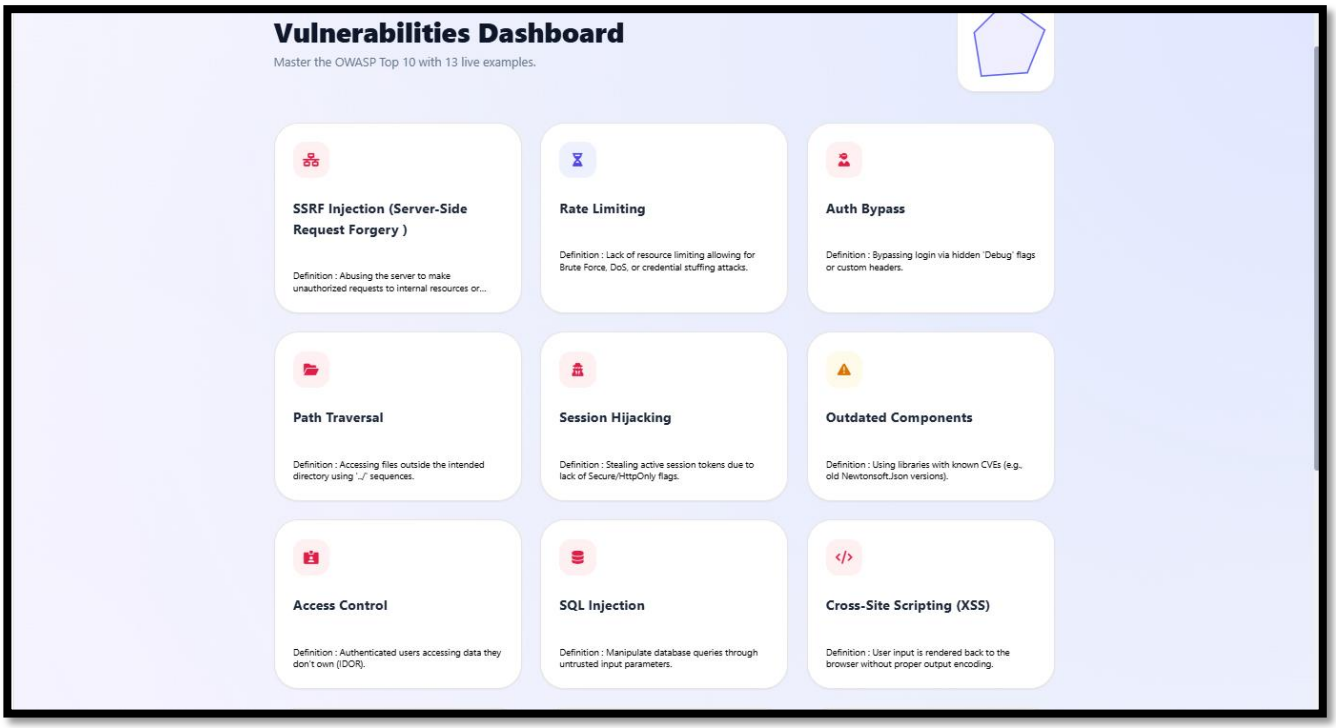
Application Security Simulation

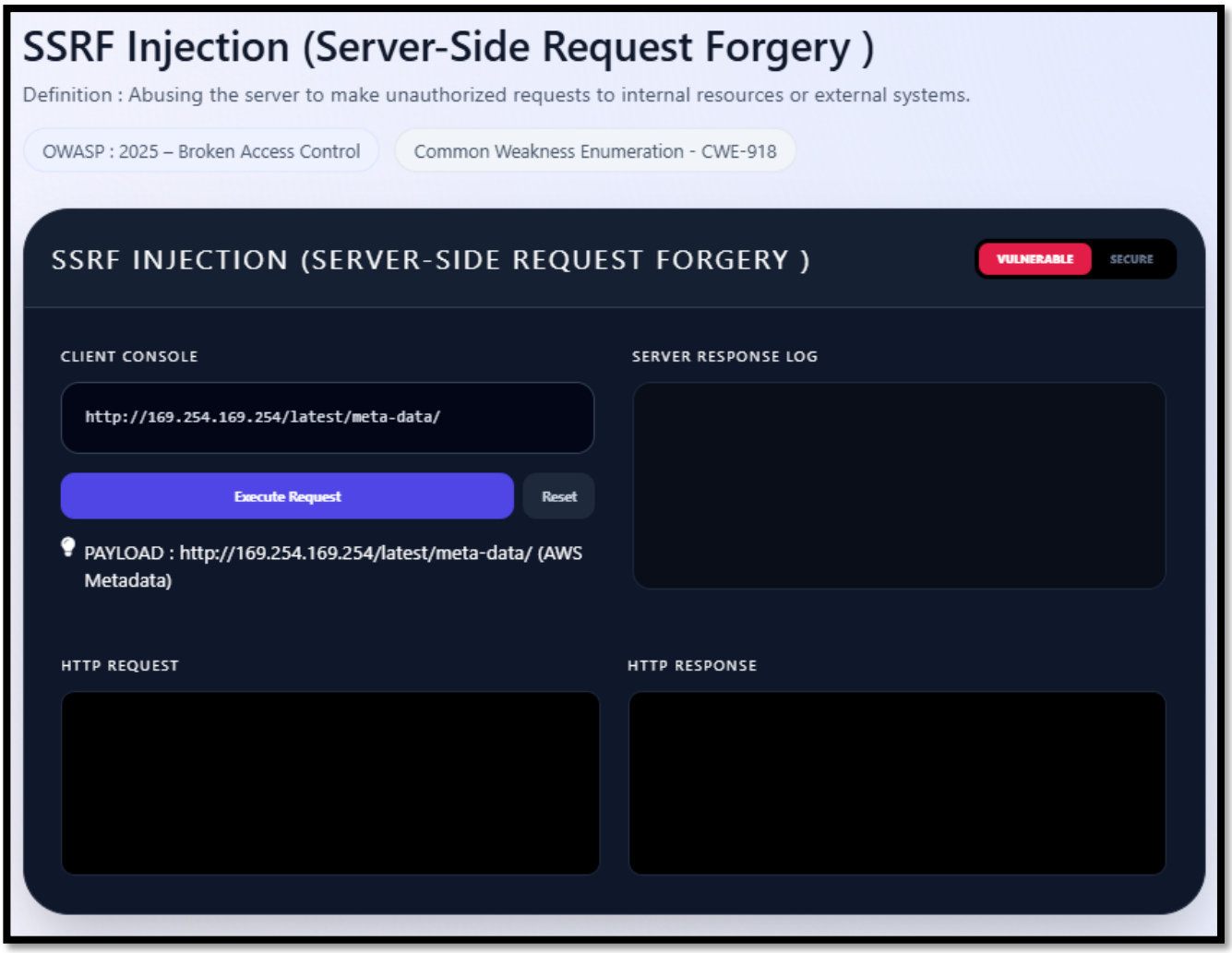
Developed by BRPL Cyber security Team

1. Open the https://harvinder300.github.io/ApplicationSecuritySimulator/ you will see

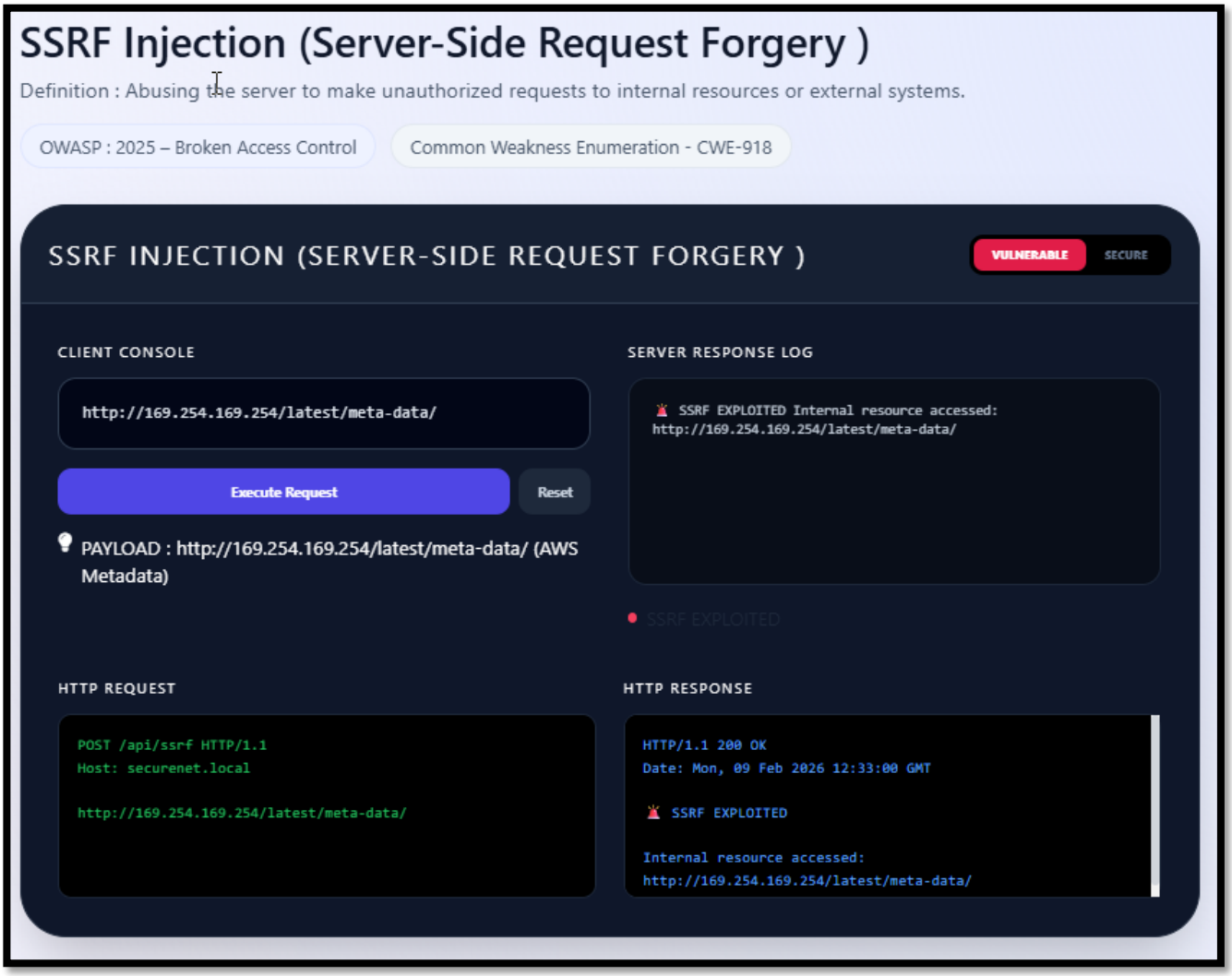
2. Click on the Launch Dashboard



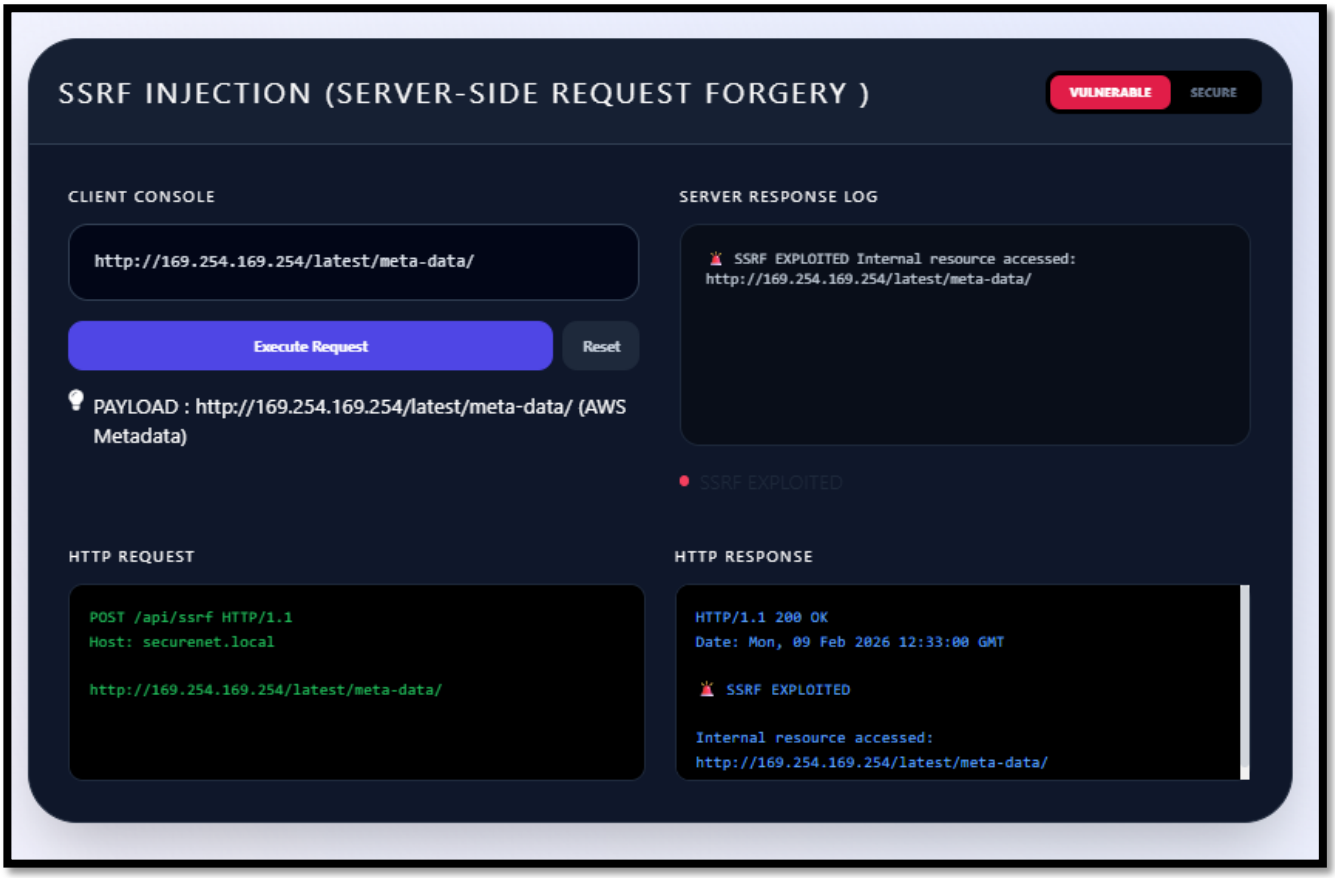
3. **Topic 1: SSRF (Server Side Request Forgery)**



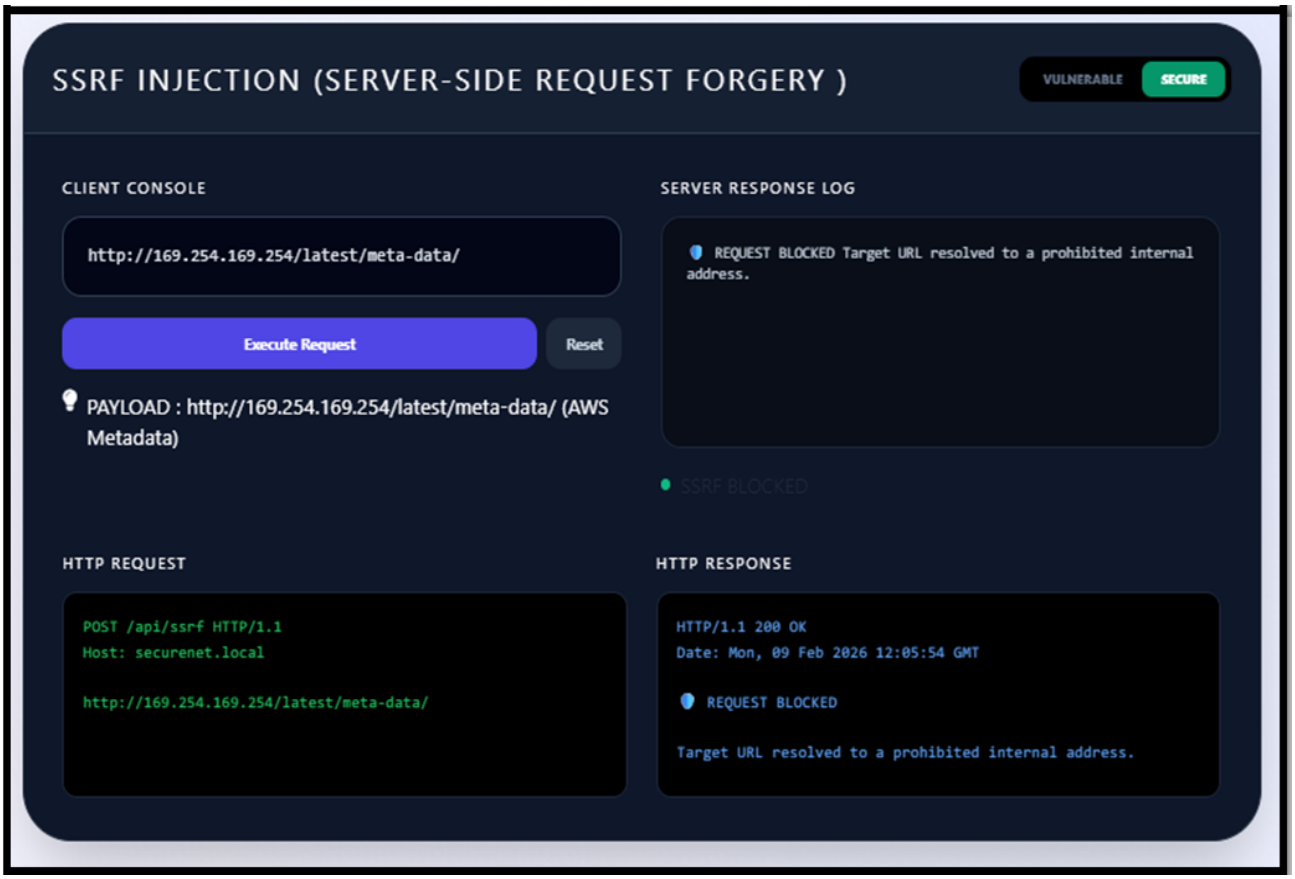
4. SSRF: Enter this payload **http://169.254.169.254/latest/meta-data/ or http://127.0.0.1 in Client Console and click Execute.**



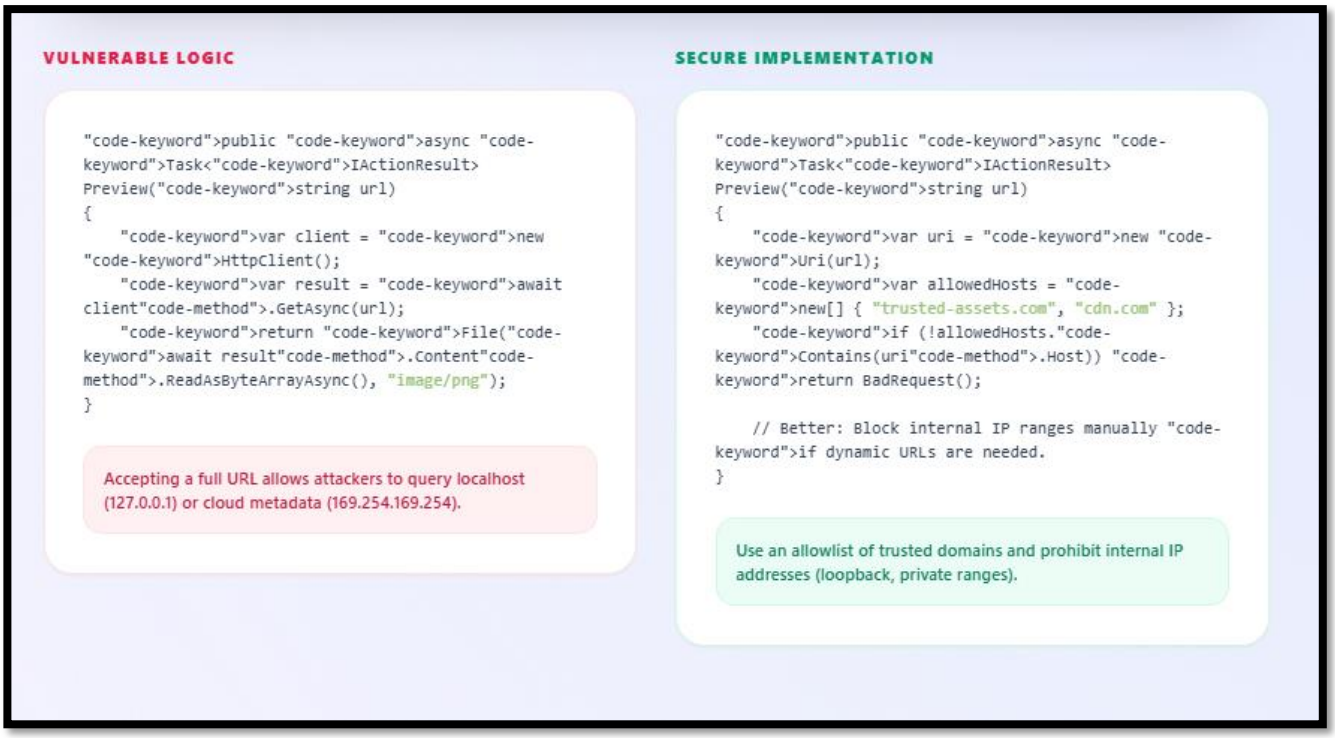
5. **SSRF: Server response against vulnerable code**



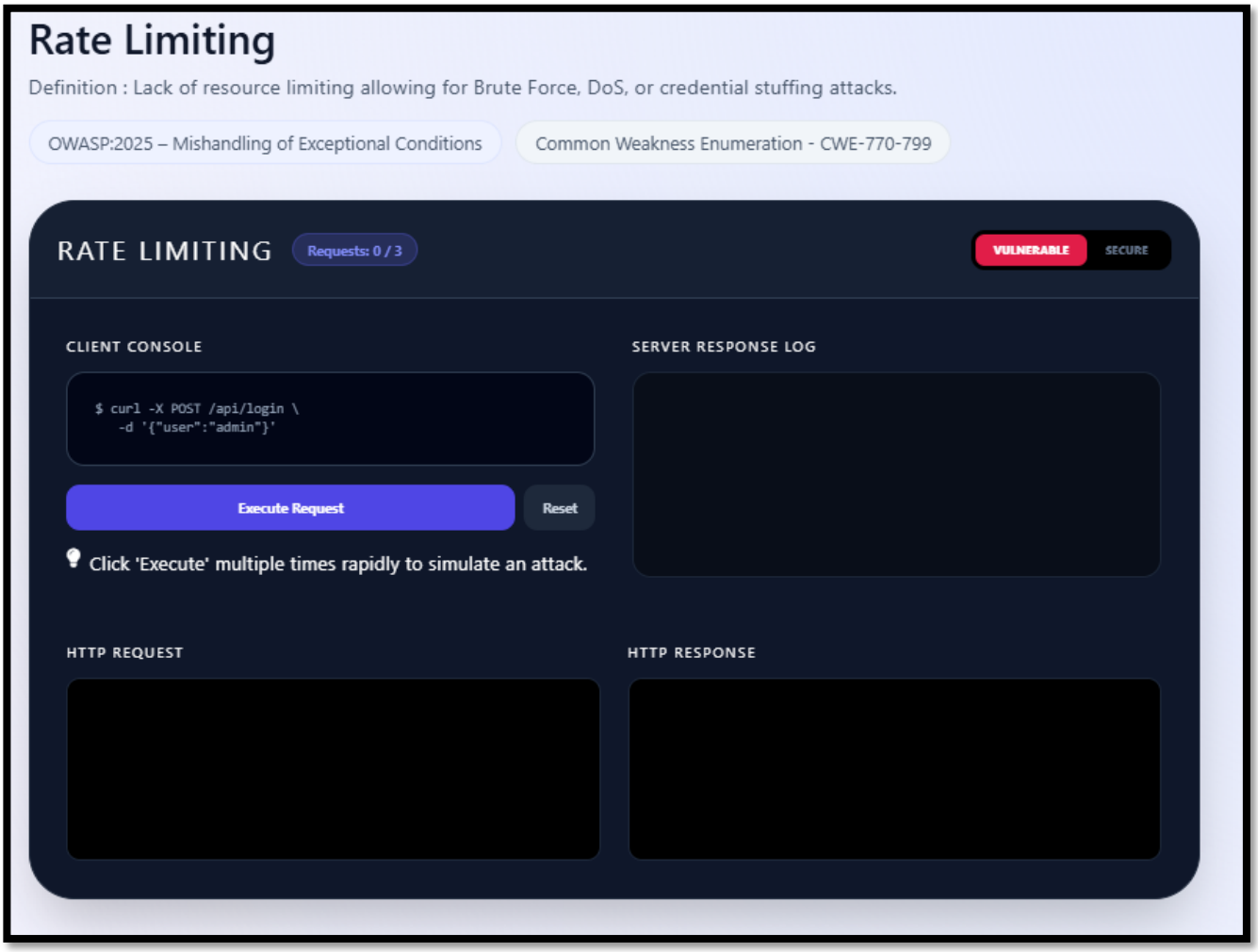
6. **SSRF:** Click on the Secure button and enter the same payload , you will see sever response when the secure code is implemented



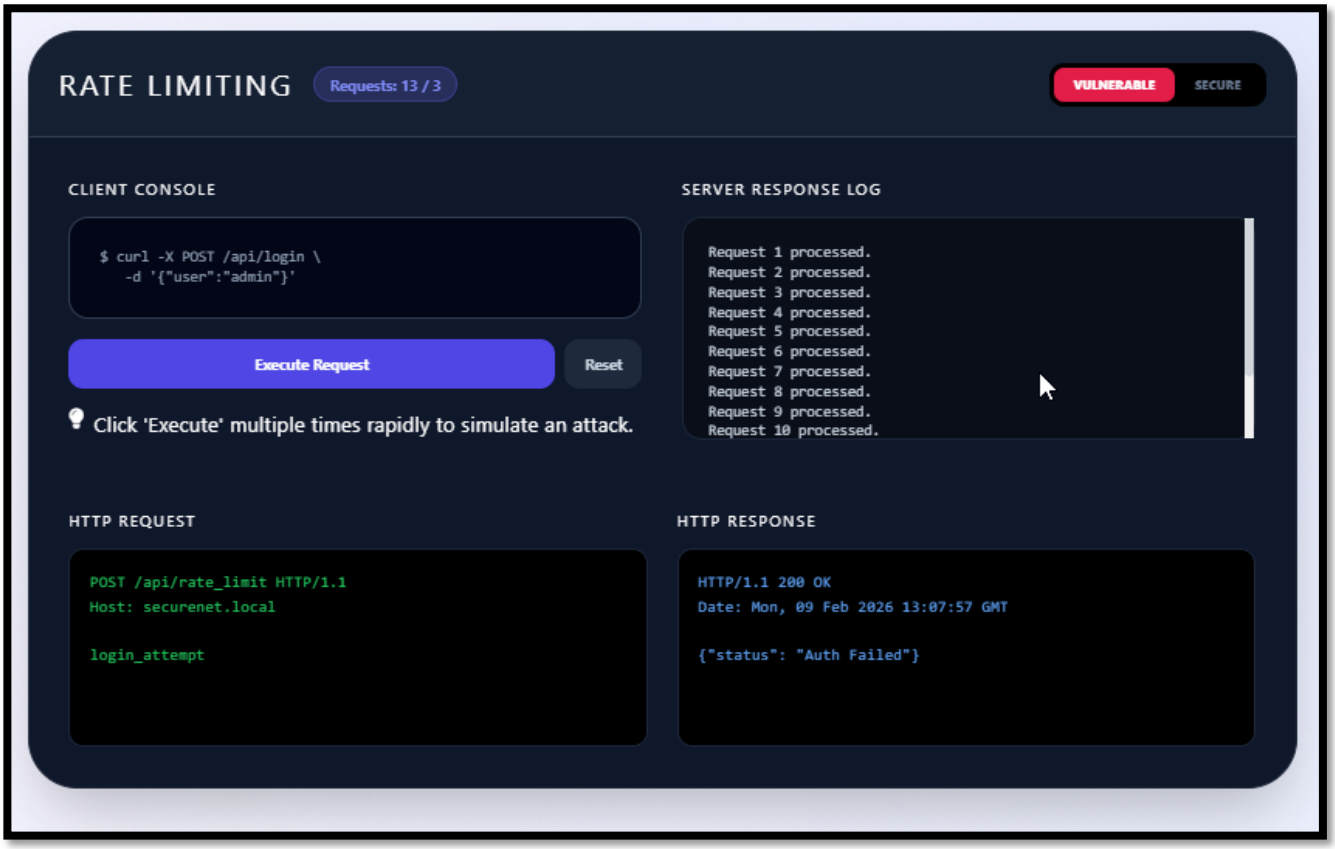
7. **Below you will see the vulnerable login and secure implementation of the code in ASP.net Core**



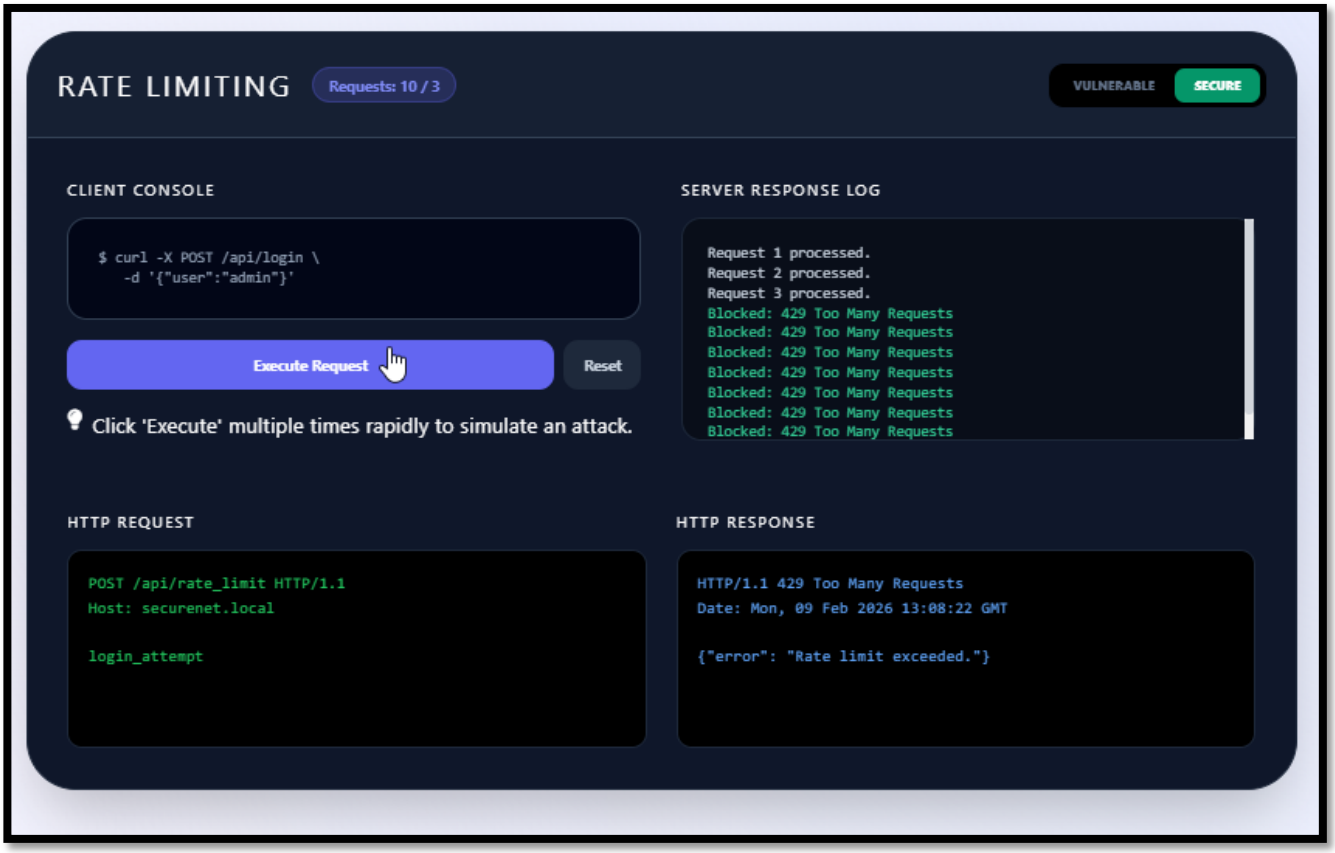
8. **Topic : Rate limiting**



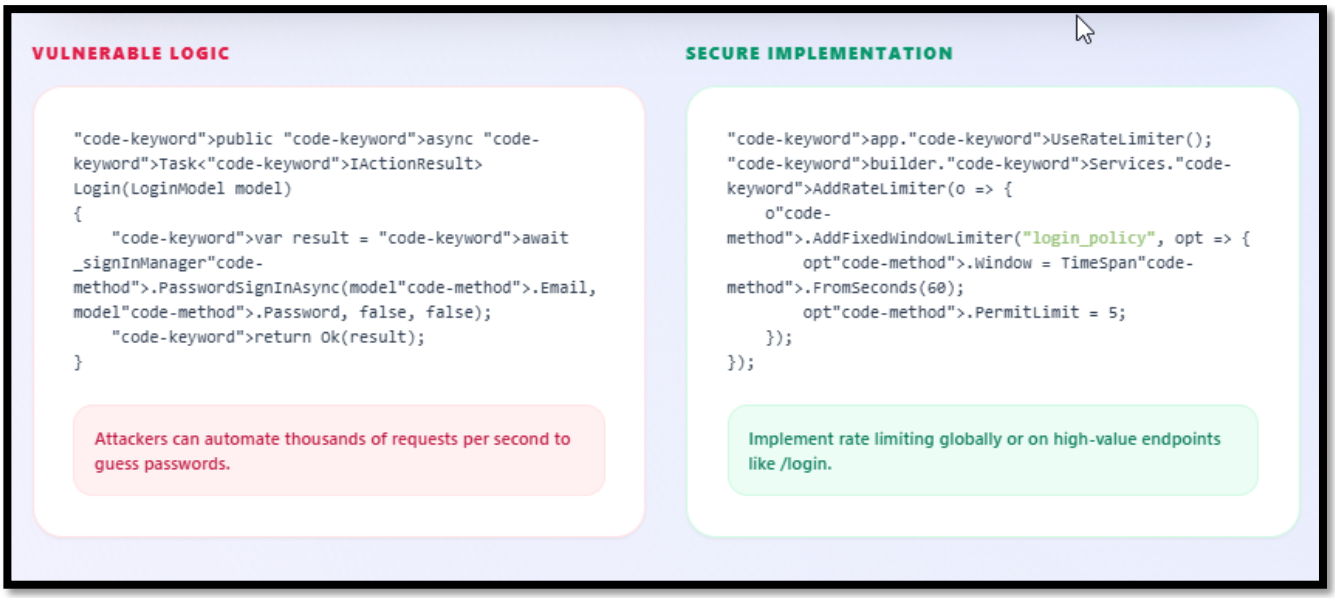
9. **Rate Limiting : If you click on execute request multiple time it will not show error , because code implementation is vulnerable to rate limiting attack**



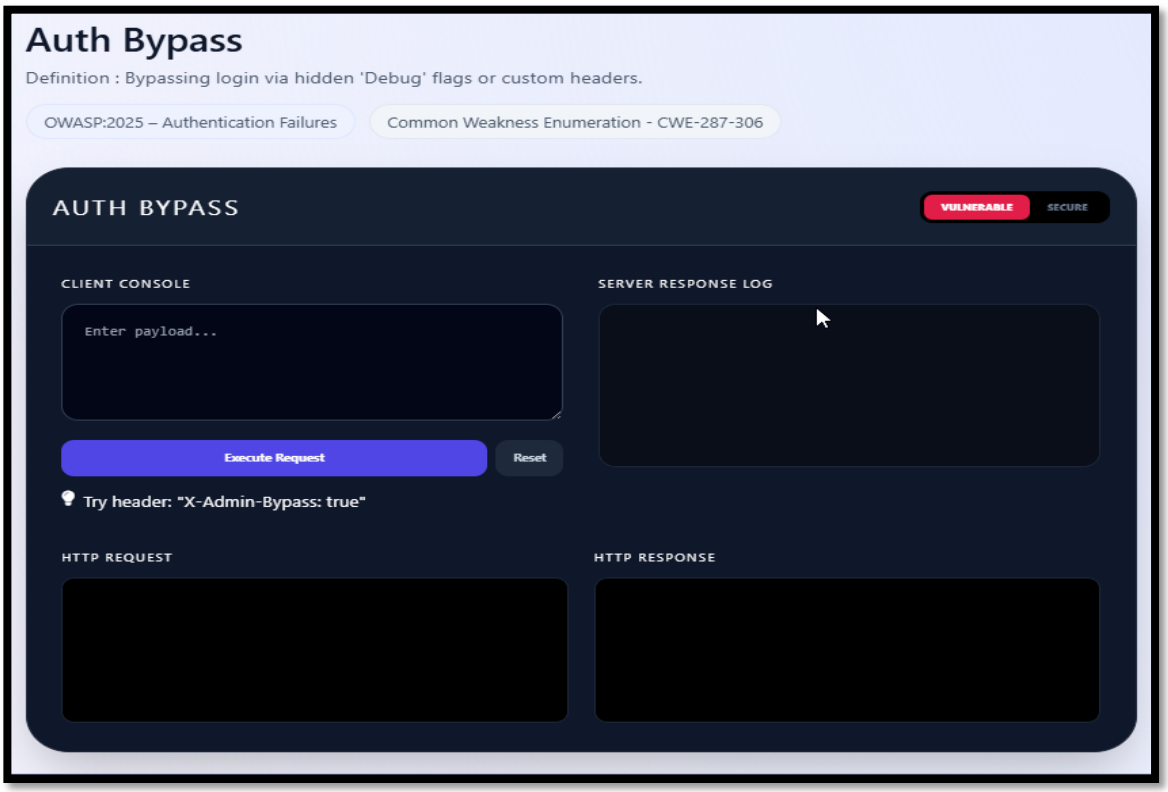
10. **Rate Limiting : If click on the Secure section and again click on execute request multiple times , it will show error of 429 too many request**



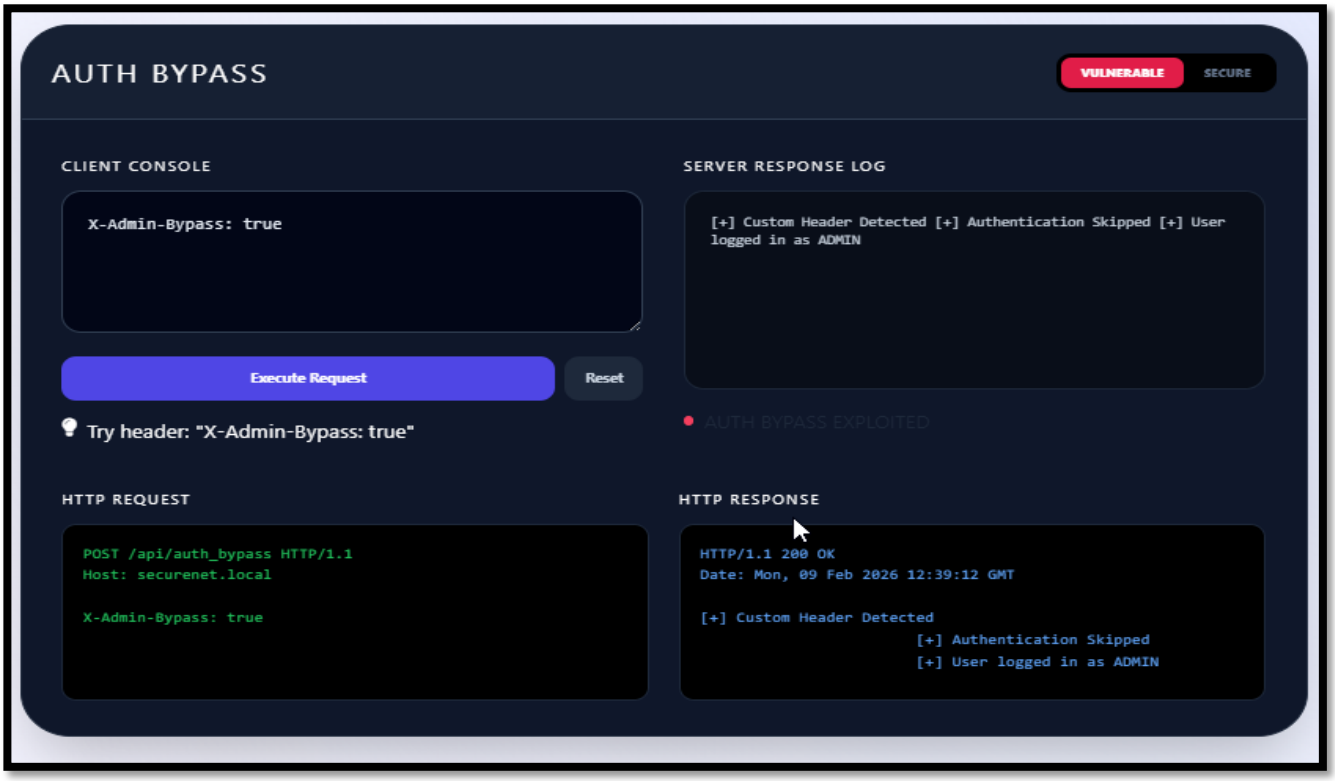
11. **Below is the code snippet of the vulnerable logic and secure implementation**



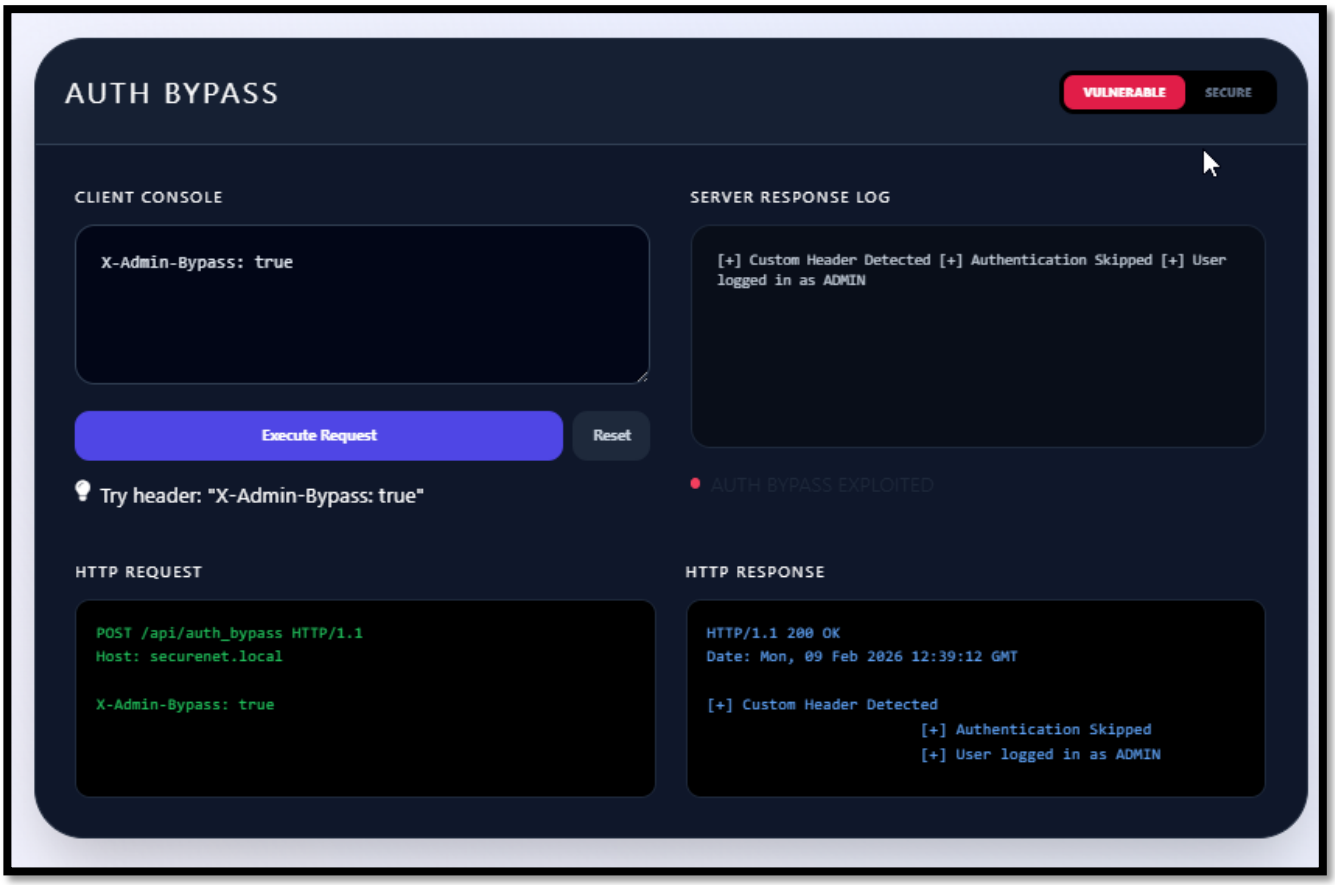
12. **Topic 3 : Auth Bypass**



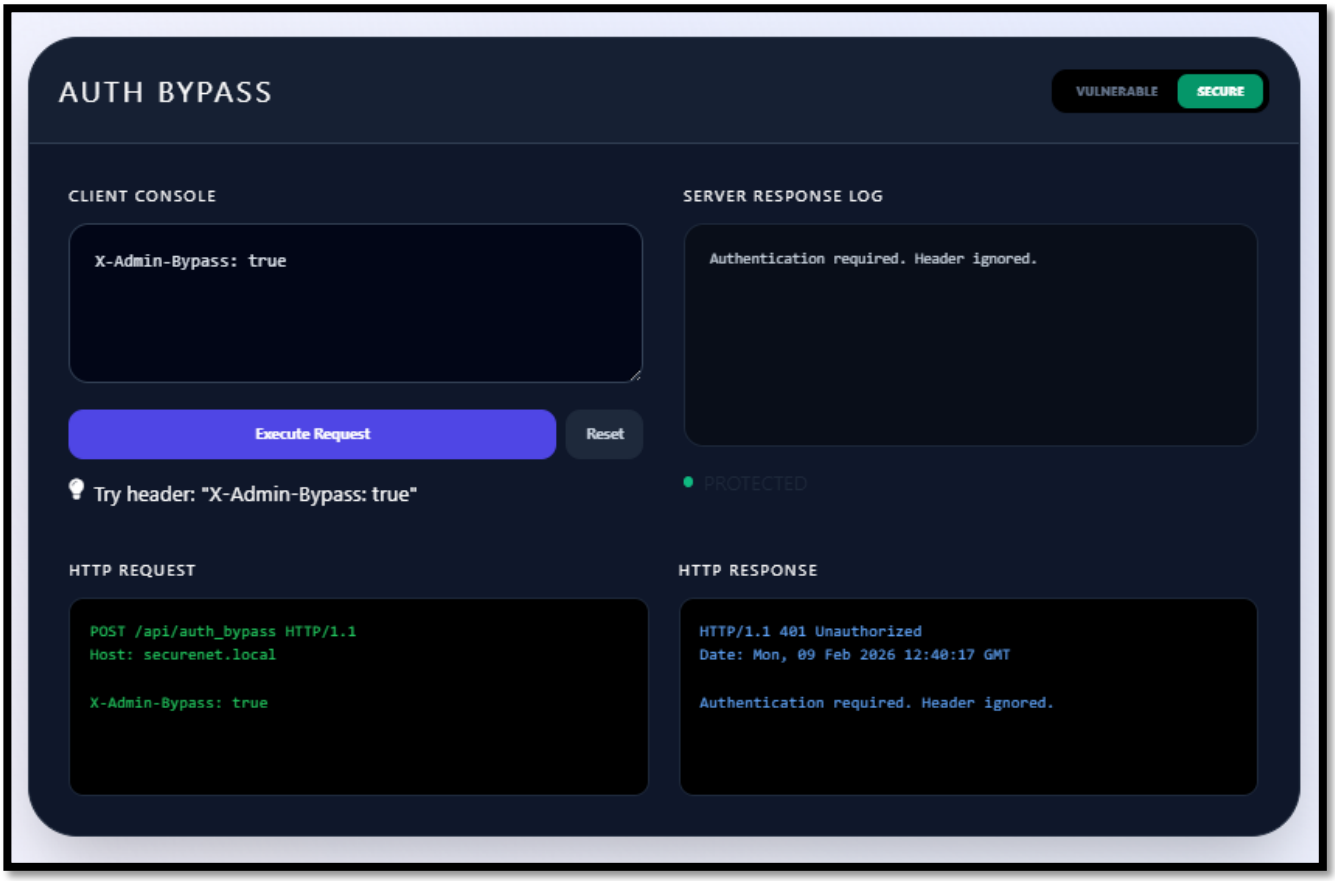
13. **Auth Bypass : Simulation for the vulnerability name authentication by response manipulation , in the input section enter this payload "X-Admin-Bypass: true"**



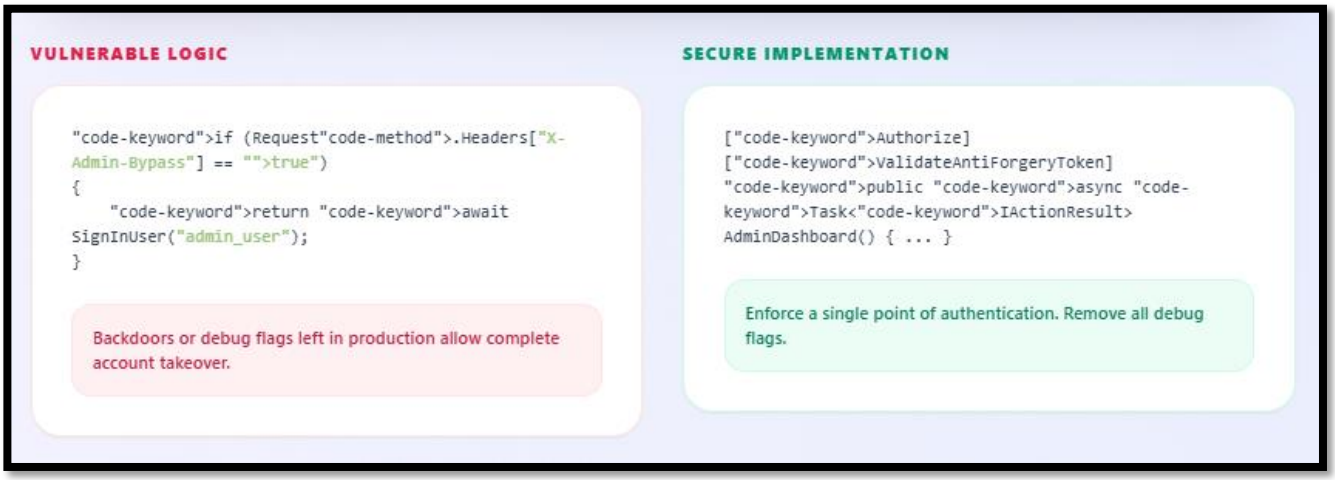
14. **Auth Bypass : You will see that authentication in bypassed by response manipulation**



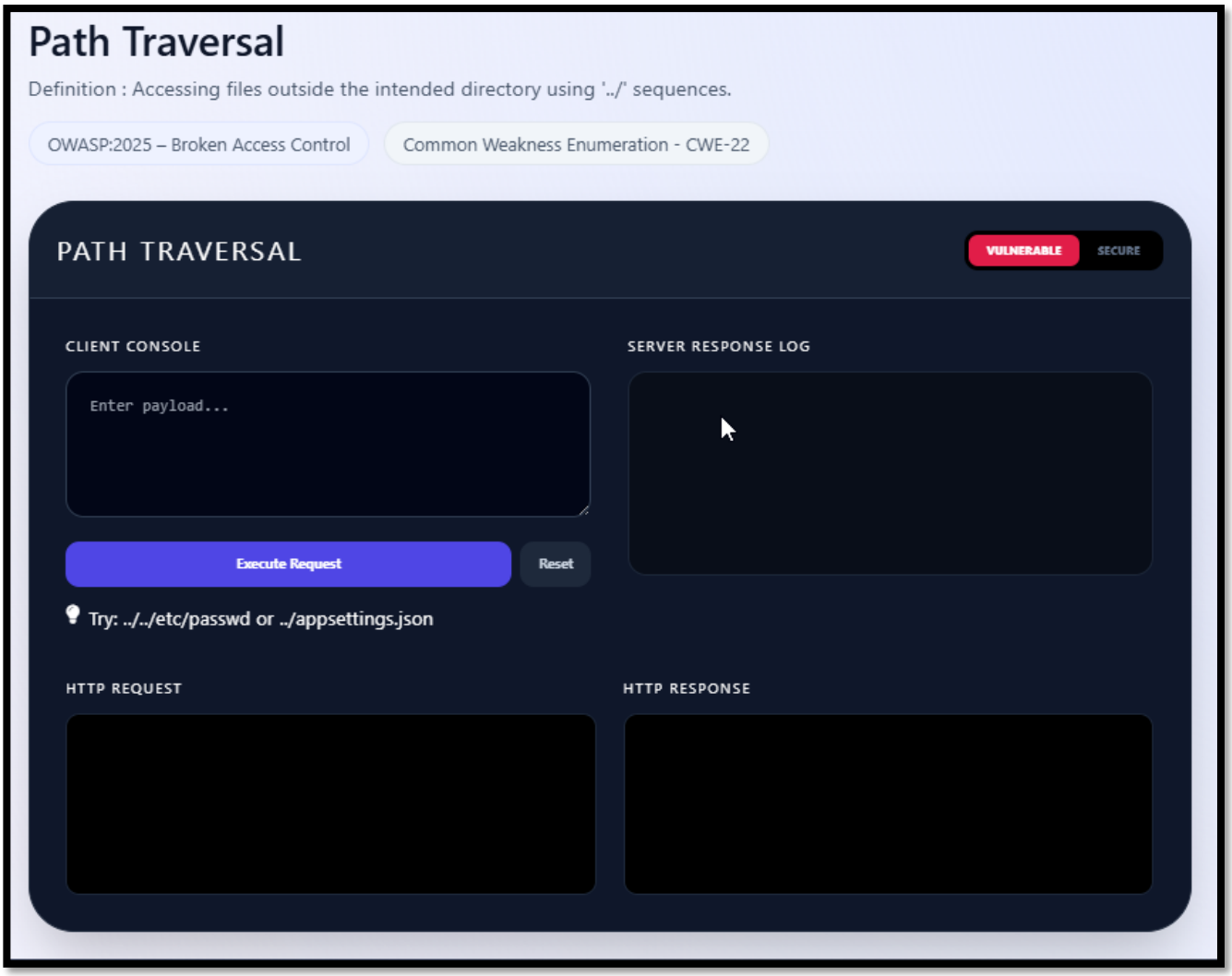
15. **Auth Bypass : When you click on the secure section , secure code is implemented so authentication is not bypass**



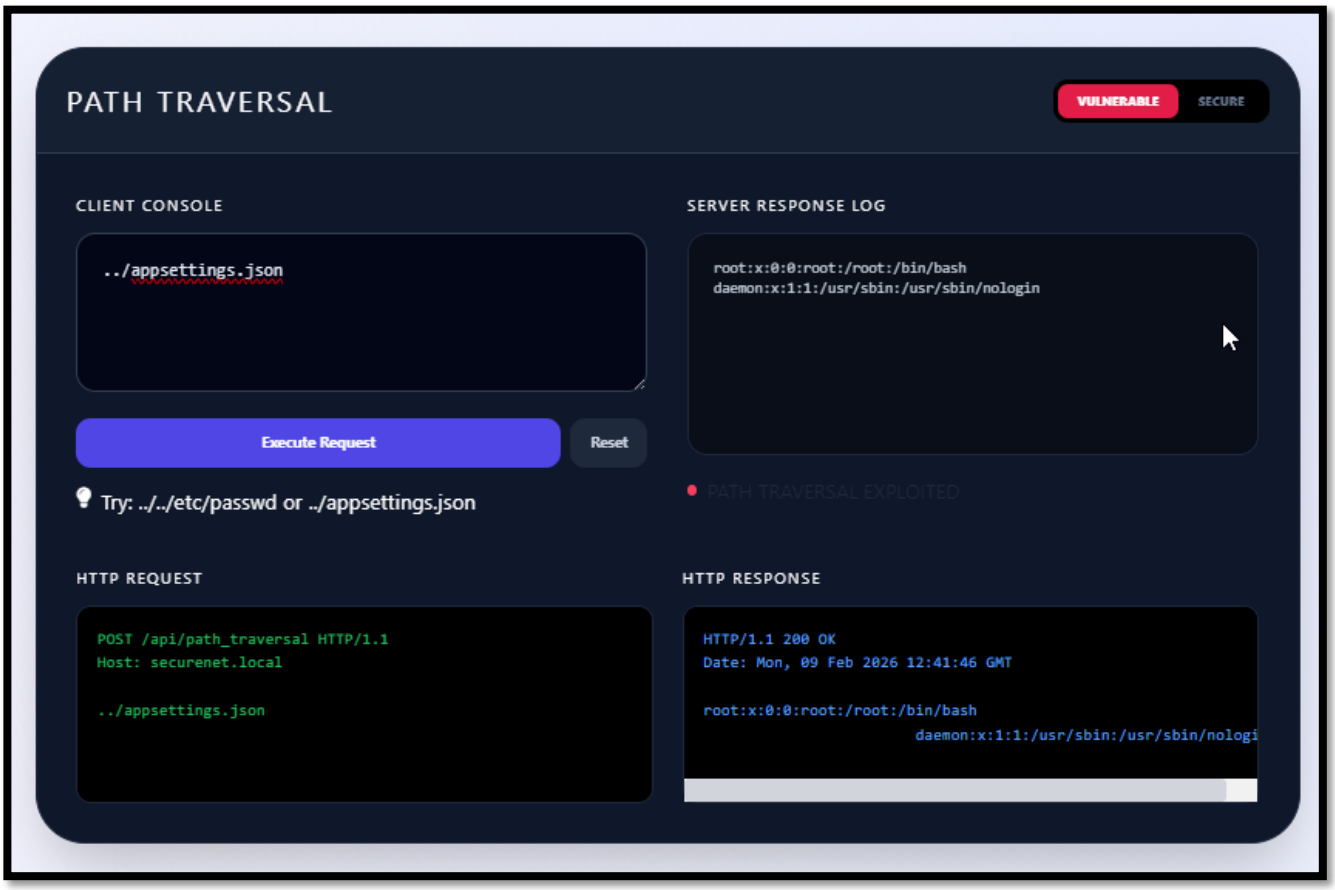
16. **Auth Bypass : Below is the vulnerable logic and secure implementation**



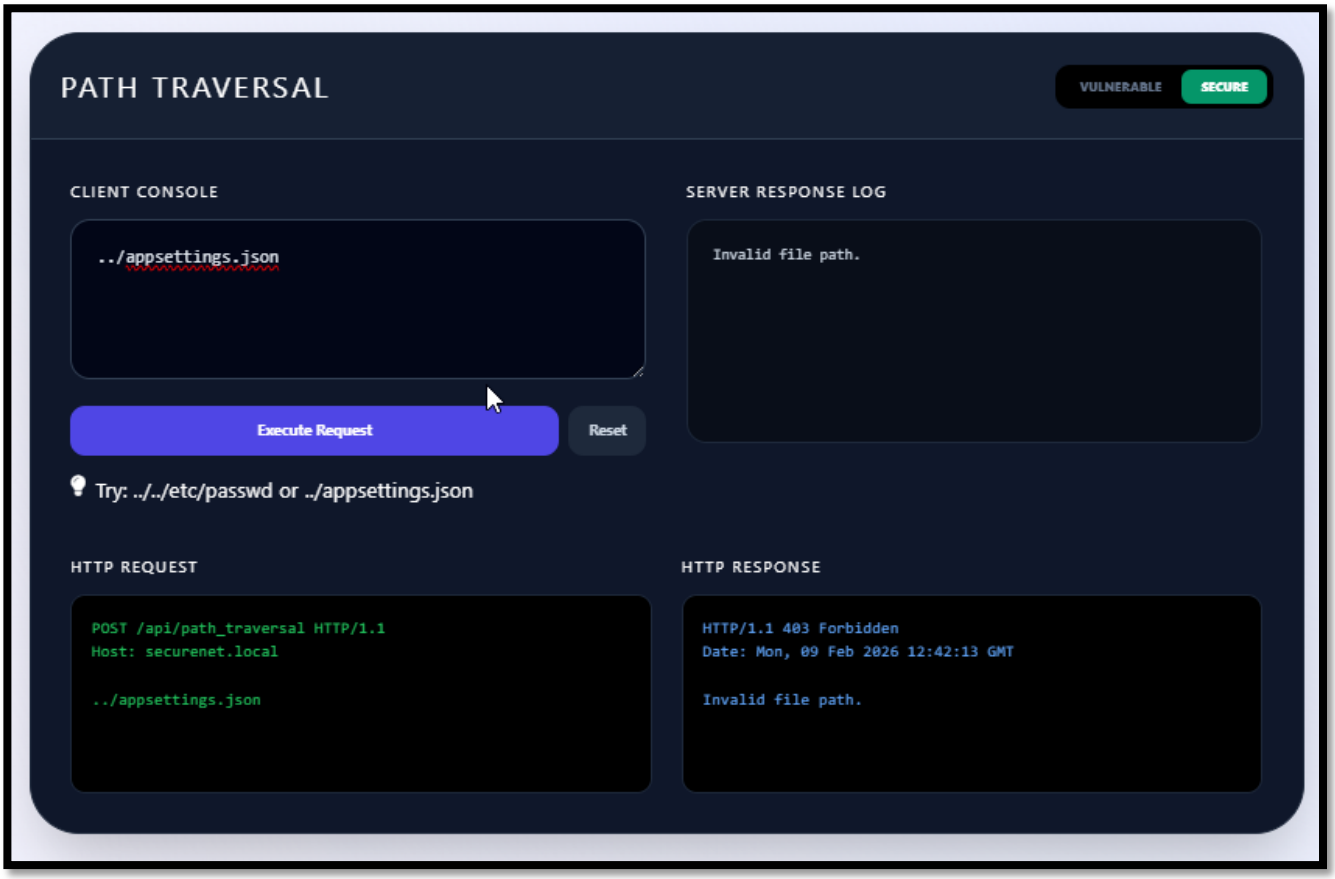
17. **Topic 4 : Path Traversal**



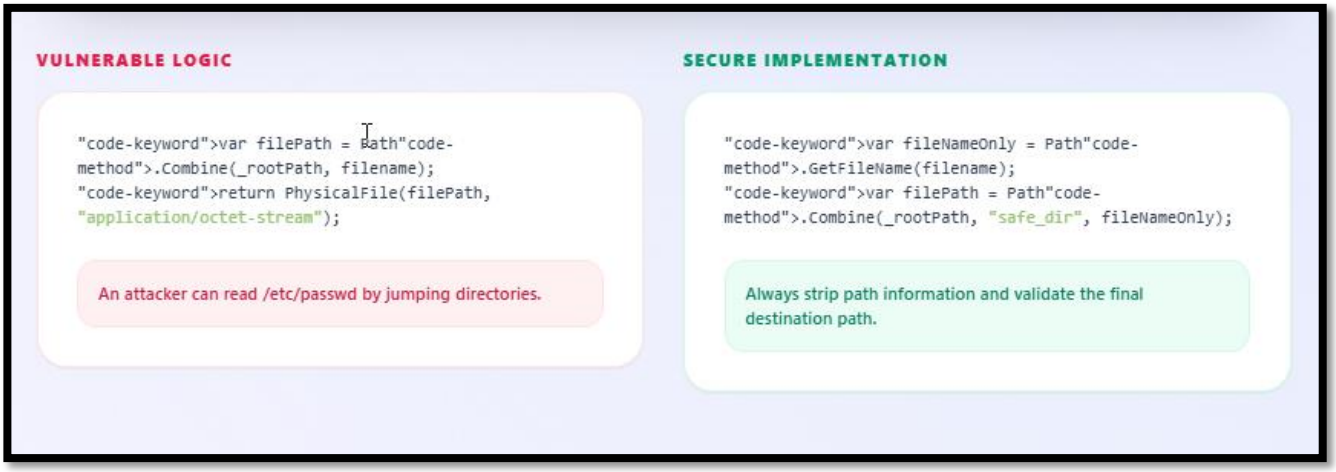
18. **Path Traversal : Enter this payload ../../etc/passwd or ../appsettings.json**



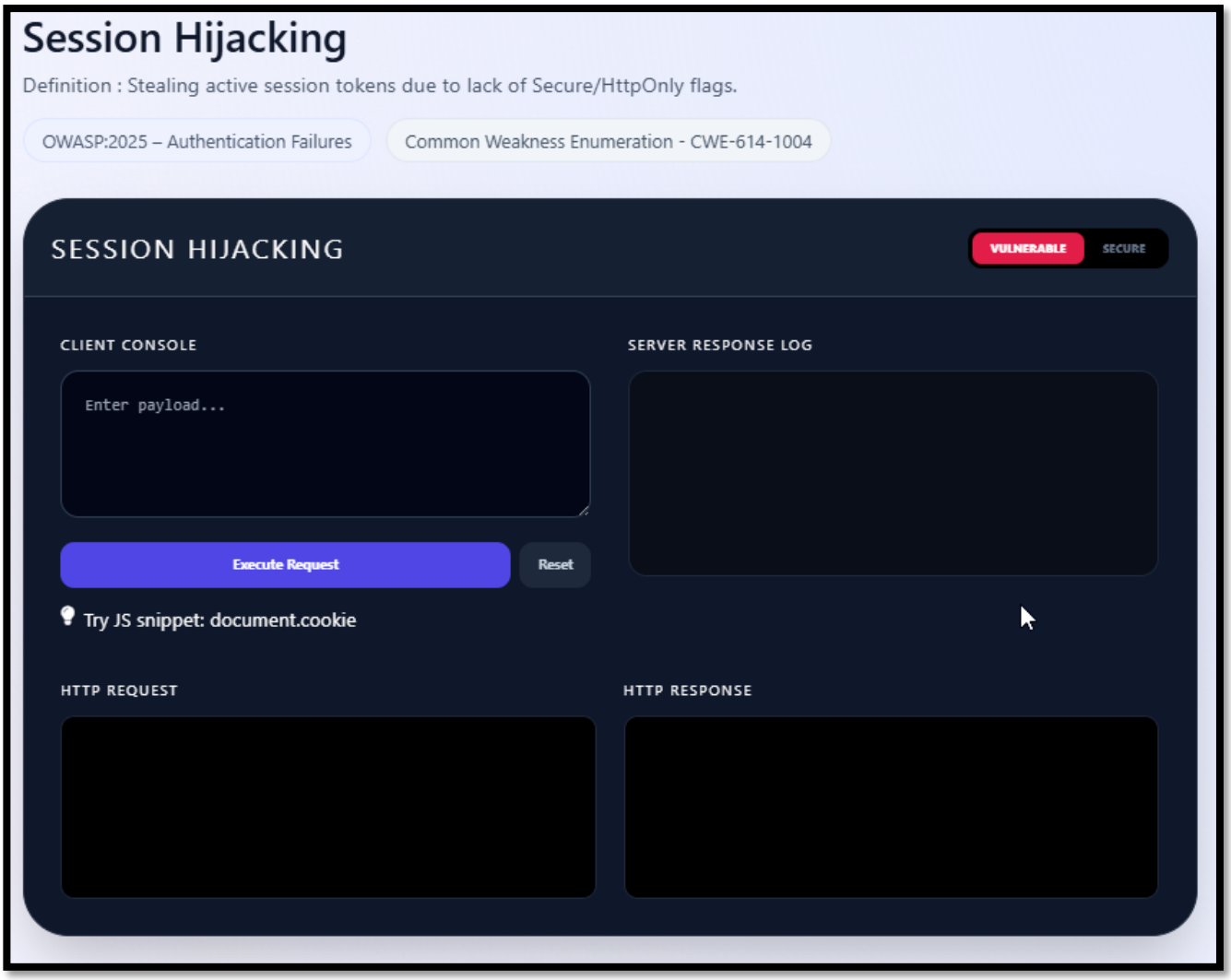
19. **Path Traversal : Click on the secure section and enter this payload again, you will see server is not giving the file.**



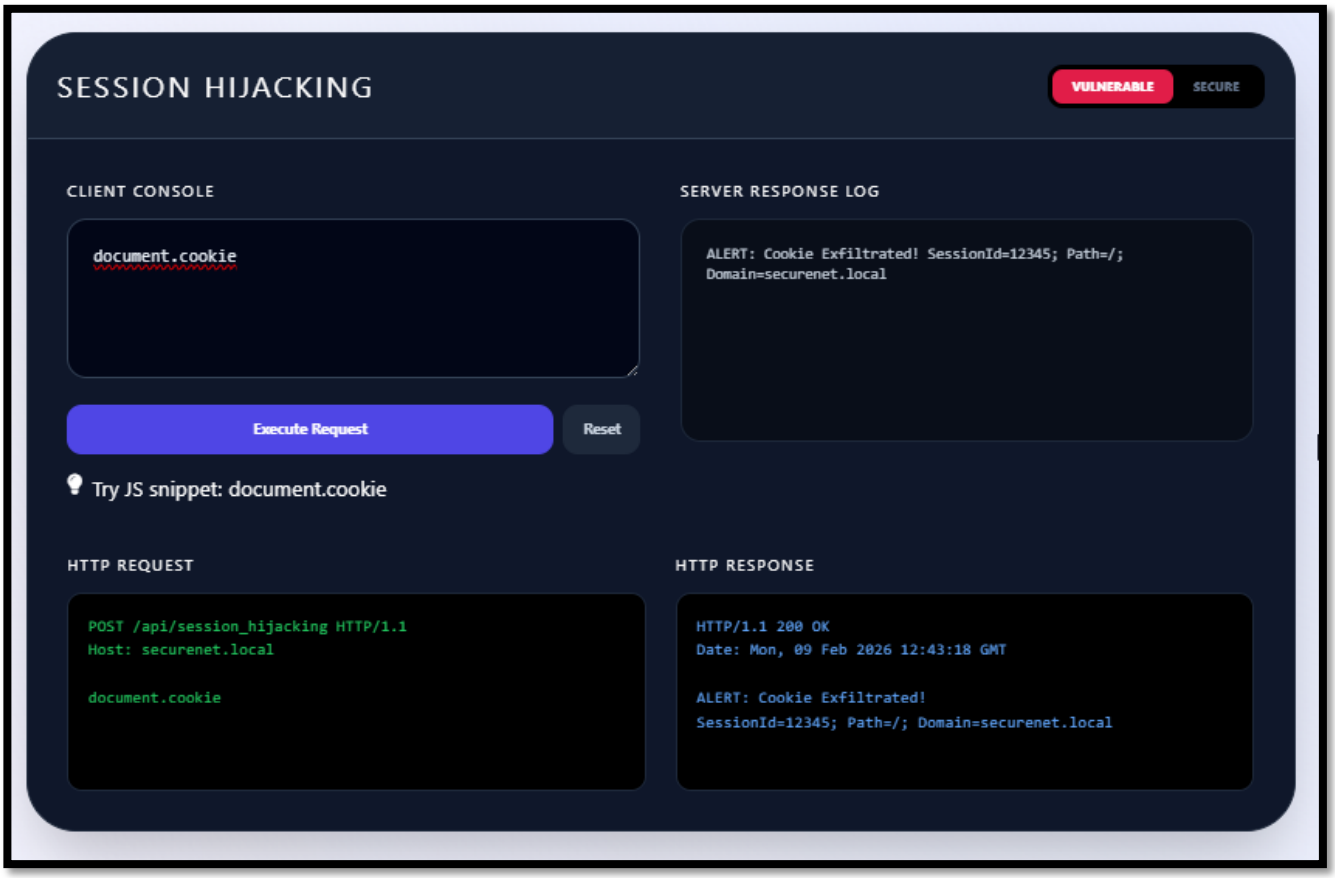
20. **Path Traversal : Below is the secure and vulnerable logic**



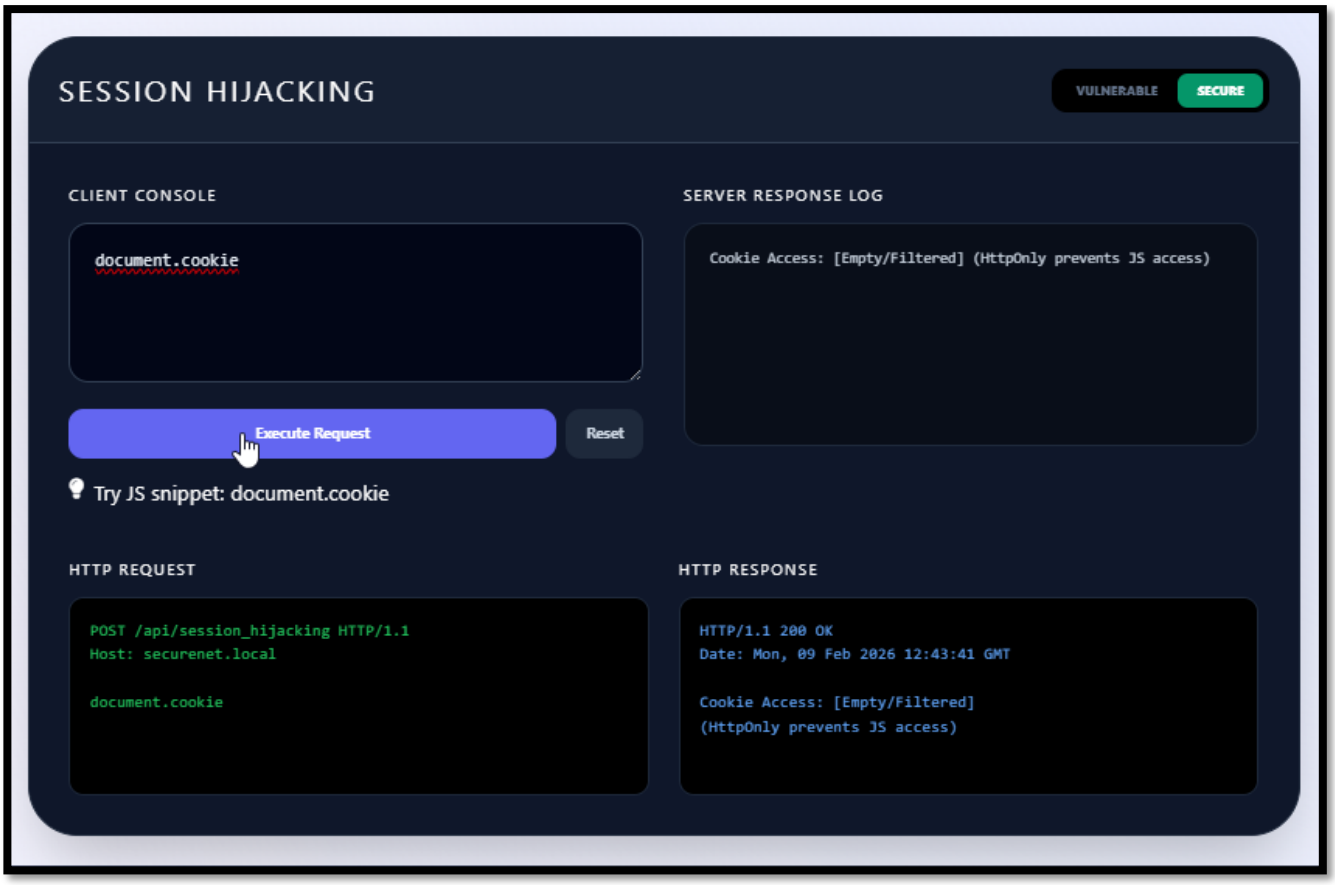
21. **Topic 5 : Session hijacking**



22. **Session hijacking :** Enter this payload **document.cookie** in the input section

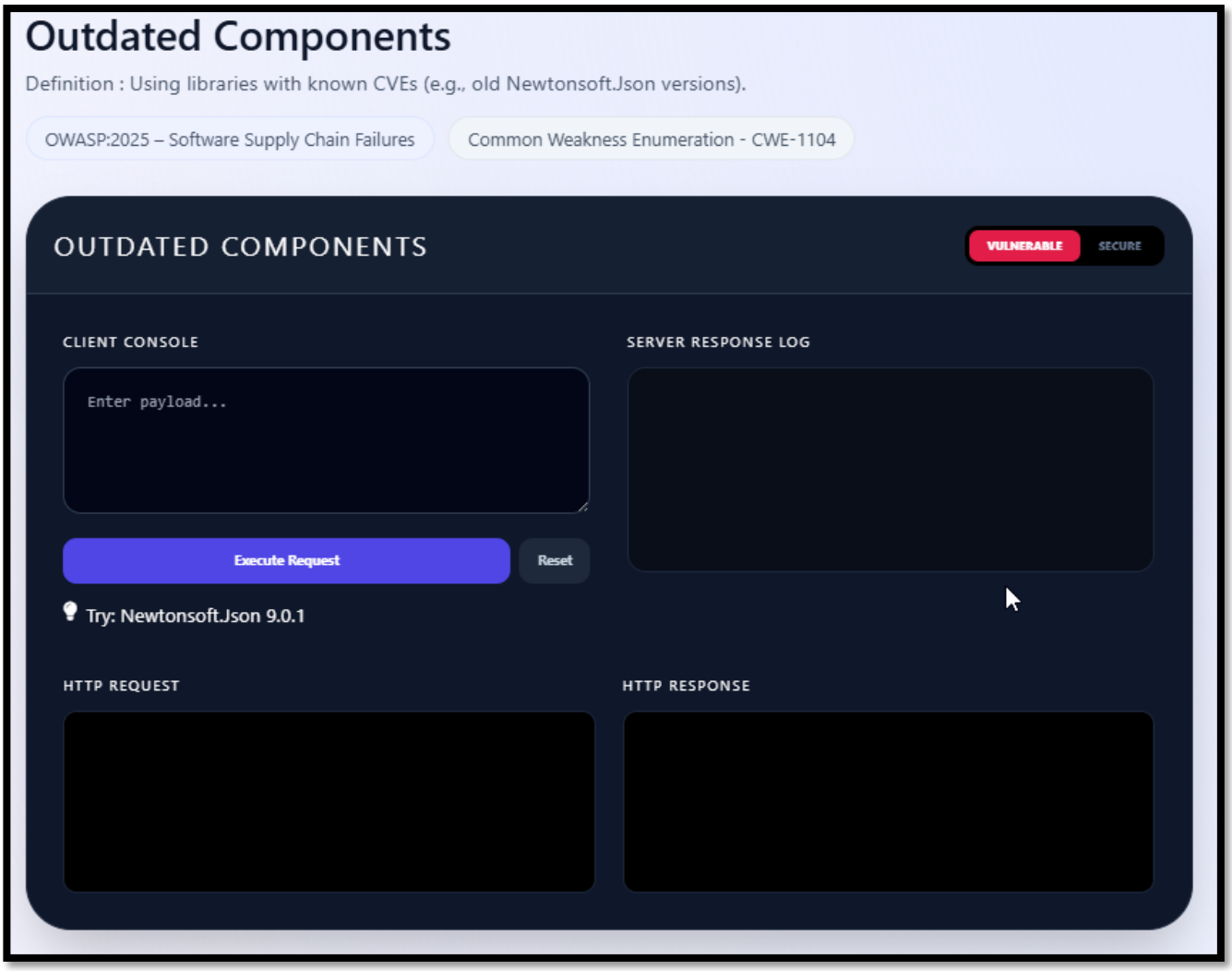


23. **Session hijacking :** Now switch to the secure mode and enter the same payload

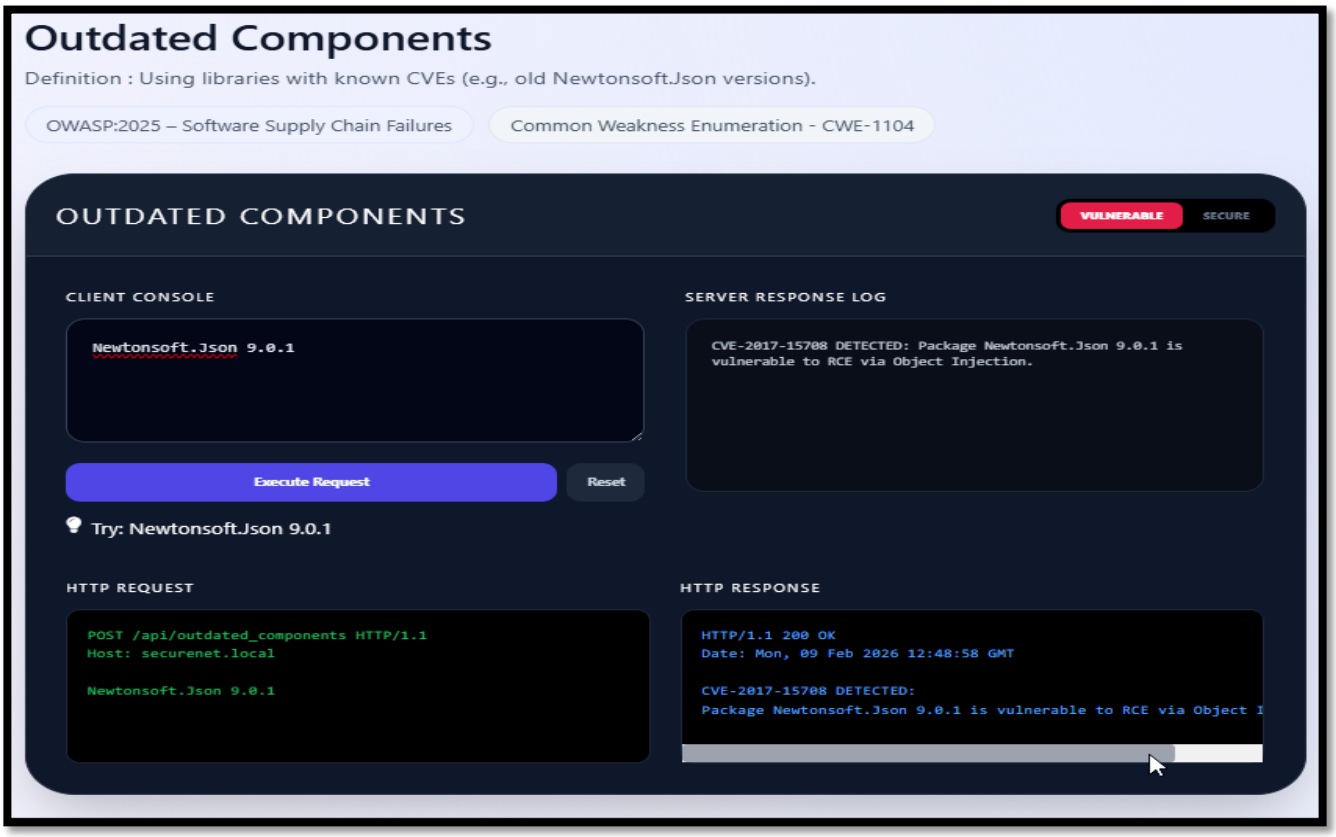


24. **Session hijacking :** Below is the secure and vulnerable logic

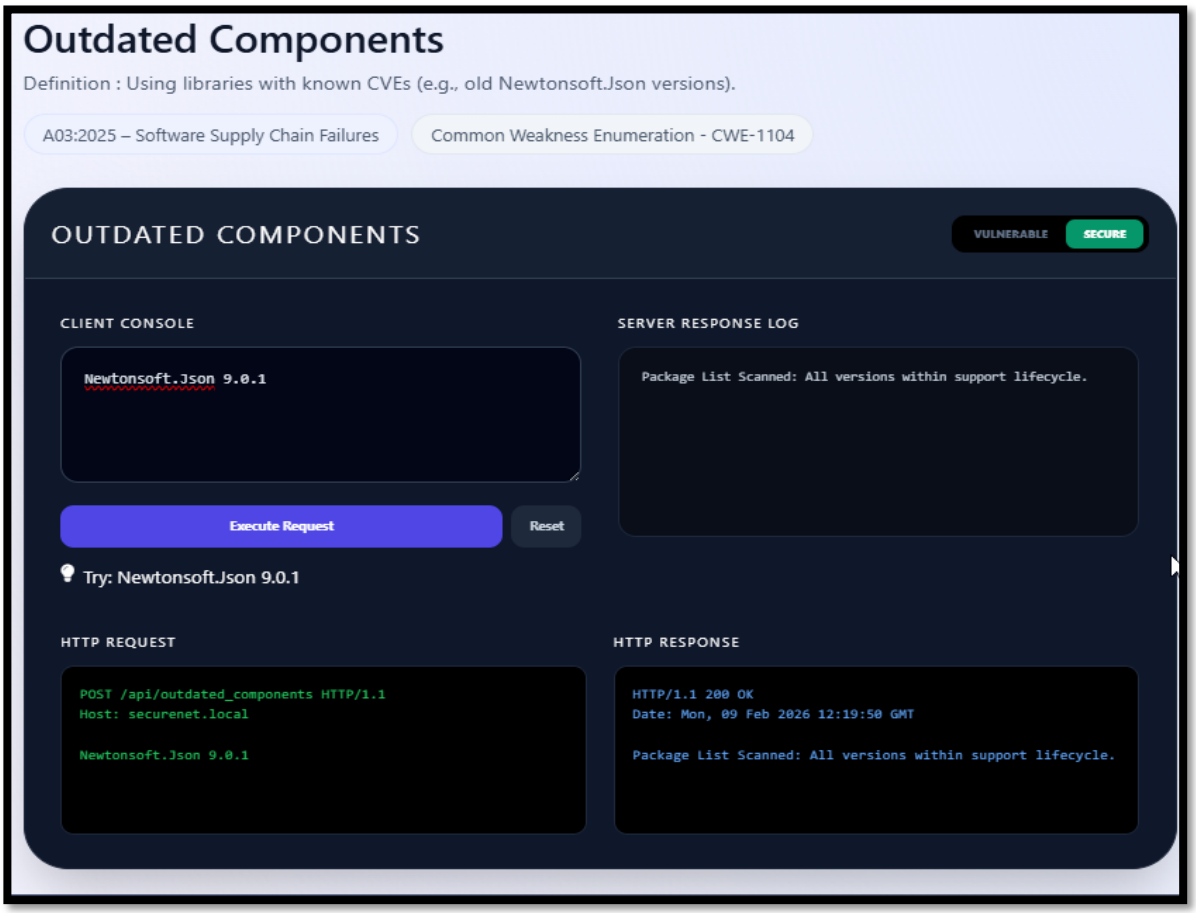
25. **Topic 6 : Outdated Components**



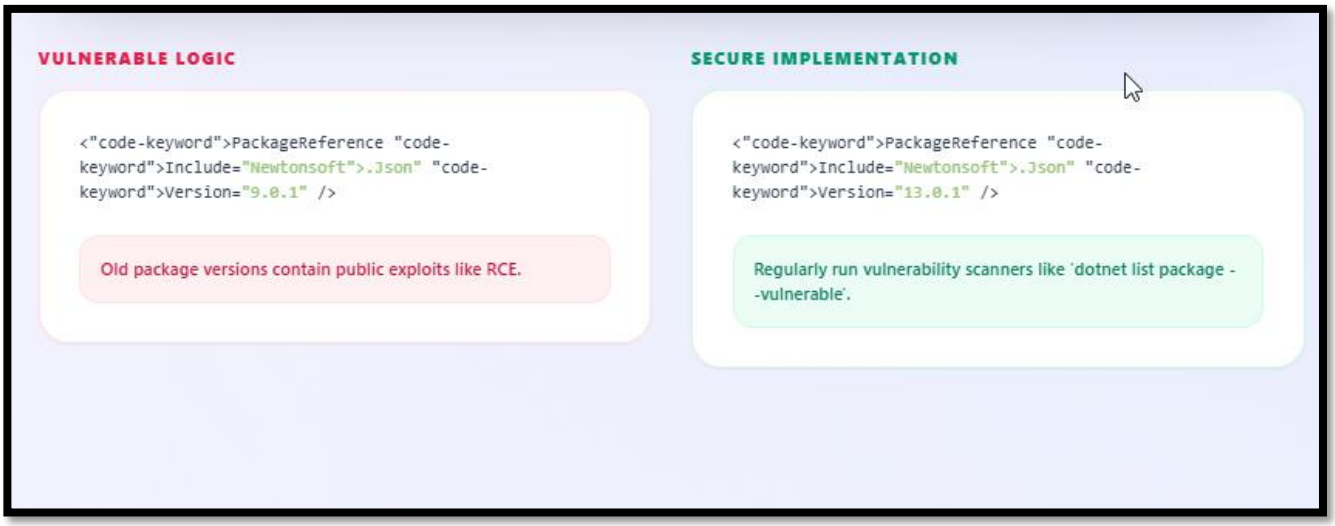
26. **Outdated Components :** Enter this payload **Newtonsoft.Json 9.0.1**



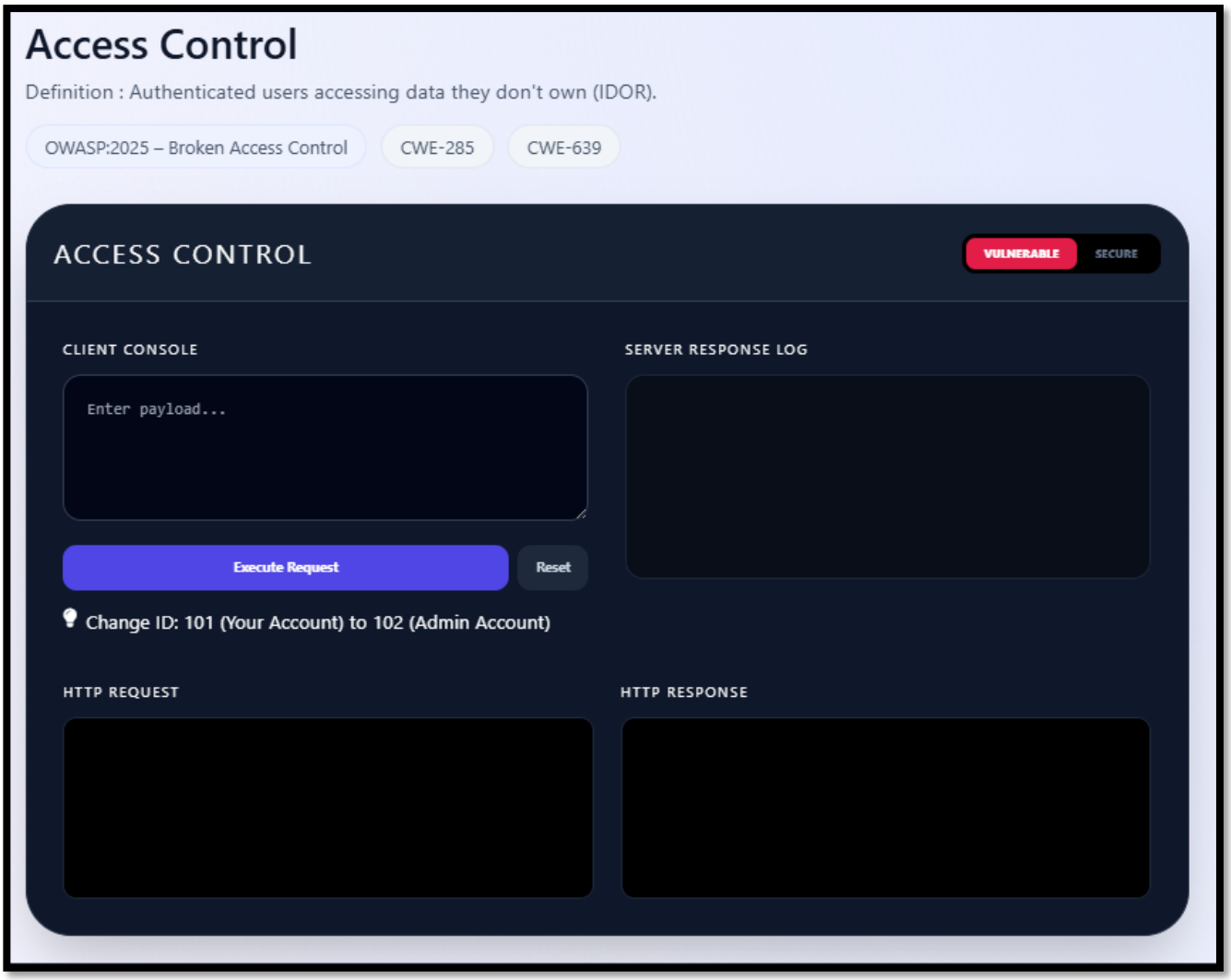
27. **Outdated Components : Now click on the Secure section and enter this payload Newtonsoft.Json 13.0.1**



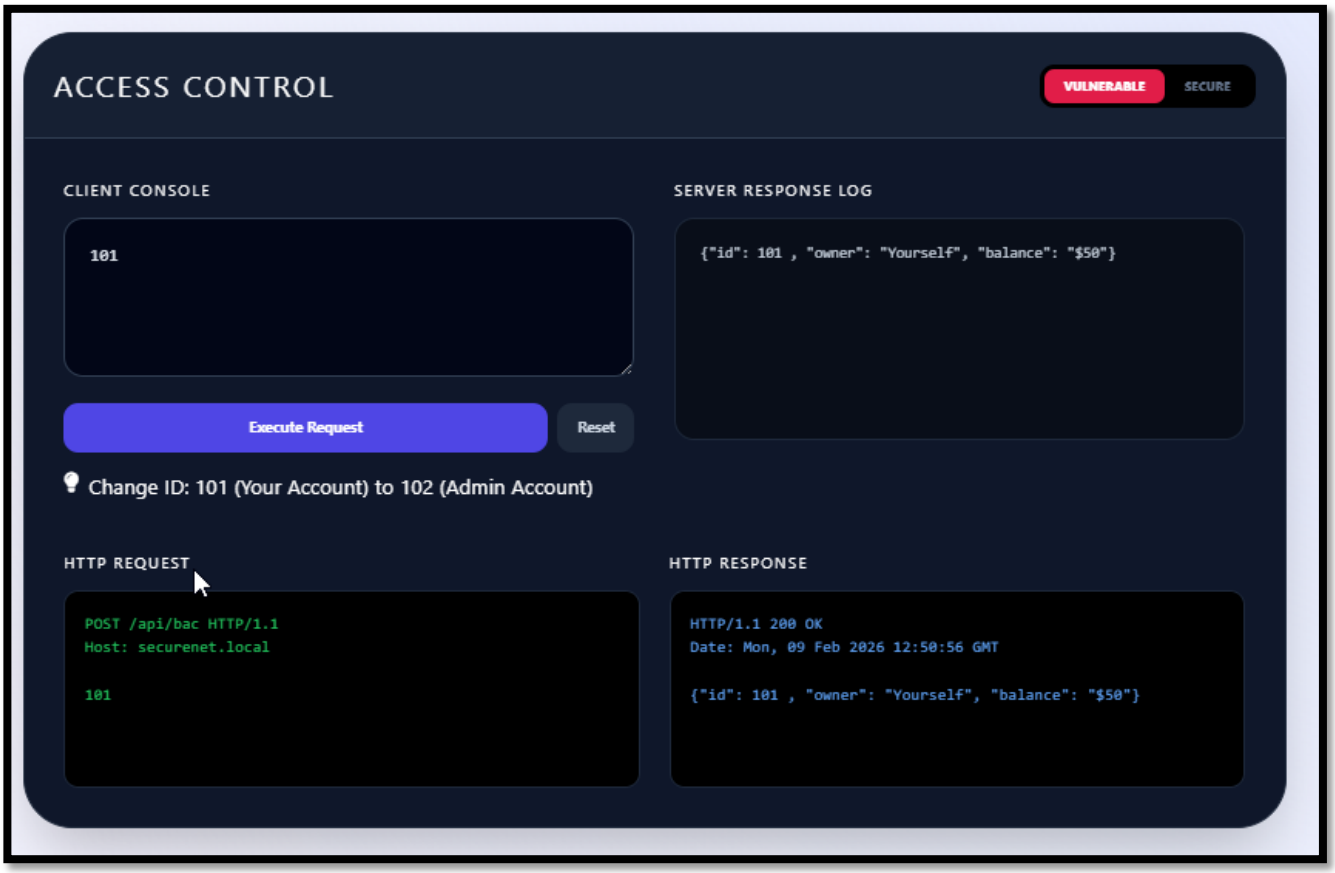
28. **Outdated Components : Below is the code for secure and vulnerable implementation**

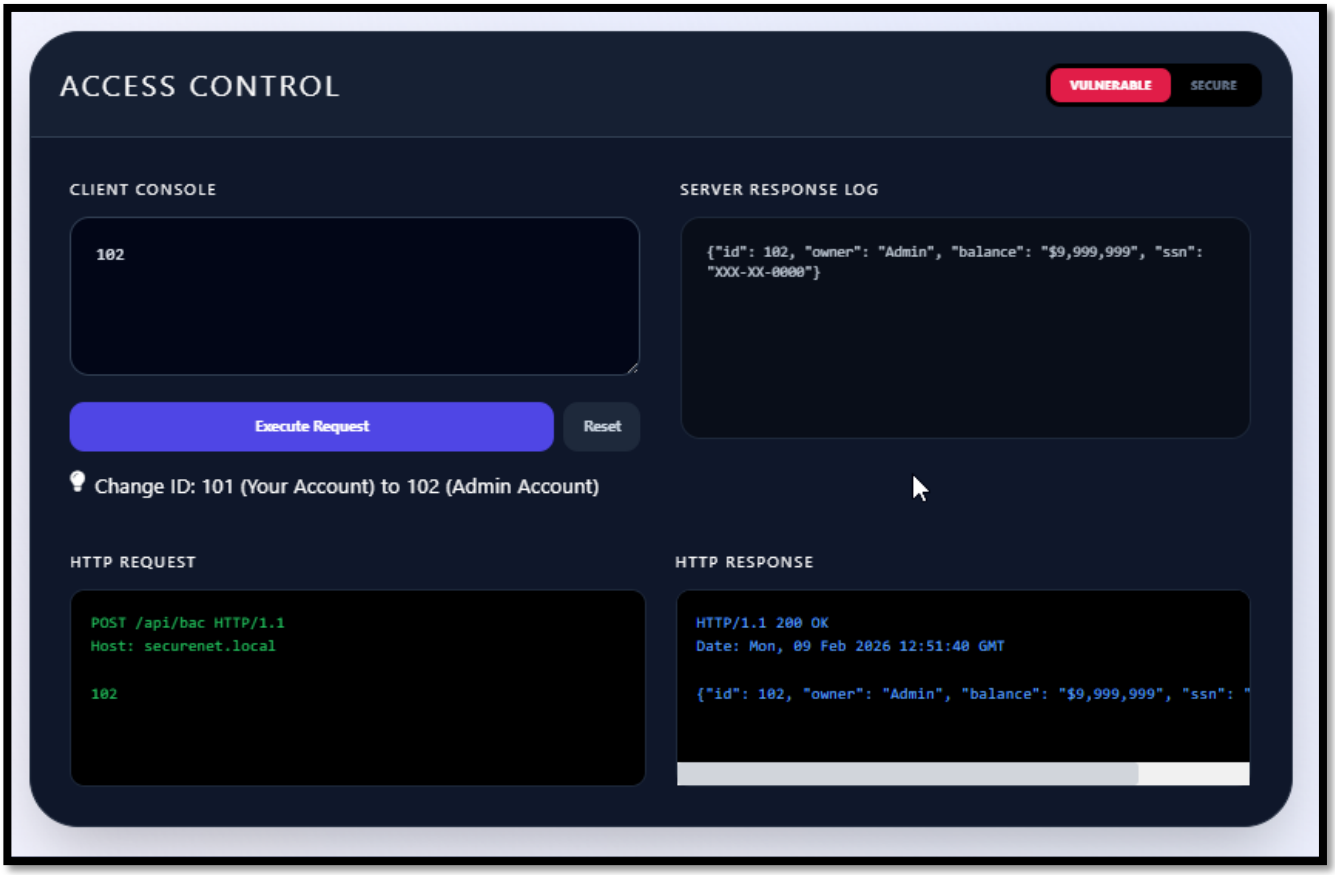


29. **Topic 7: Access Control**

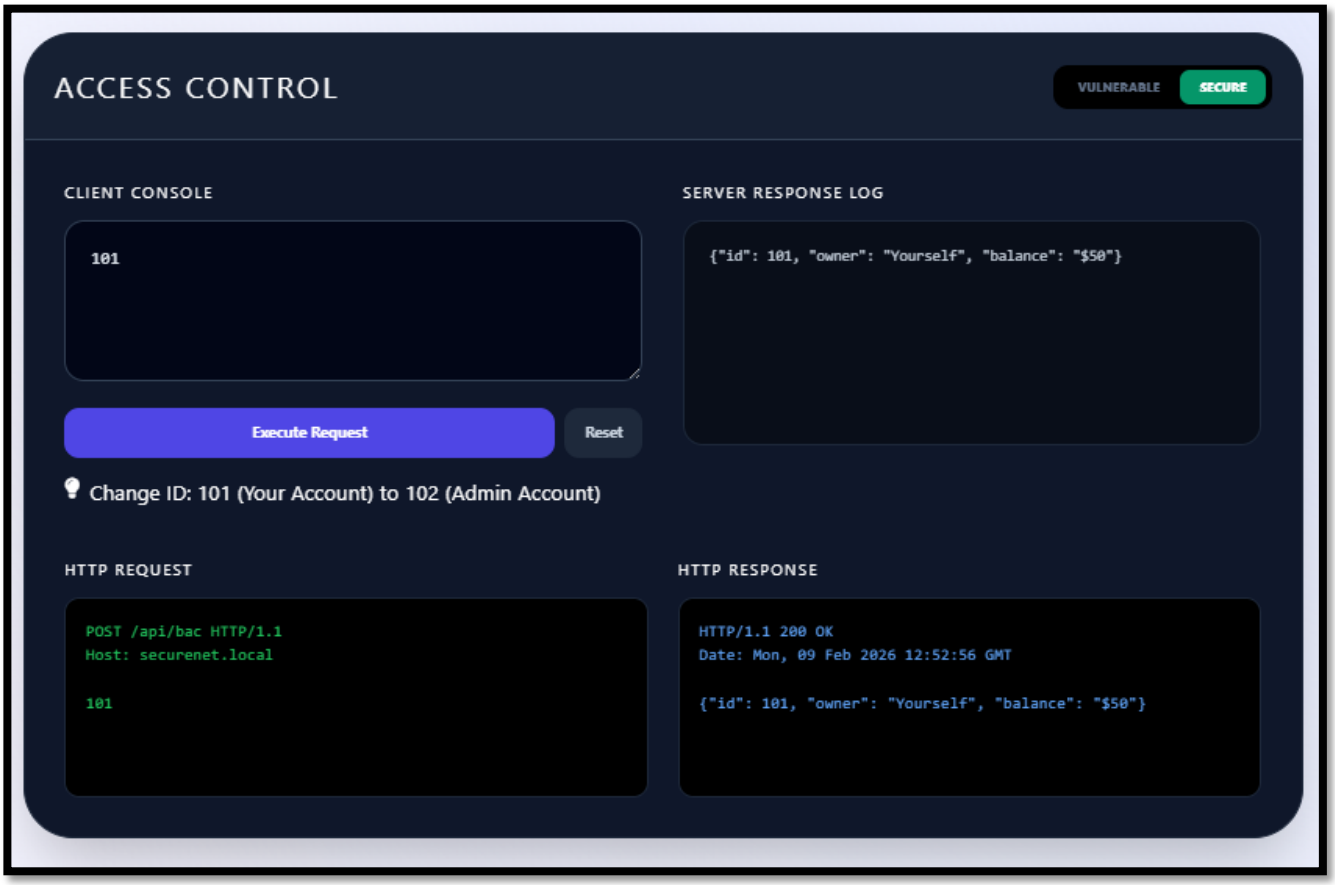


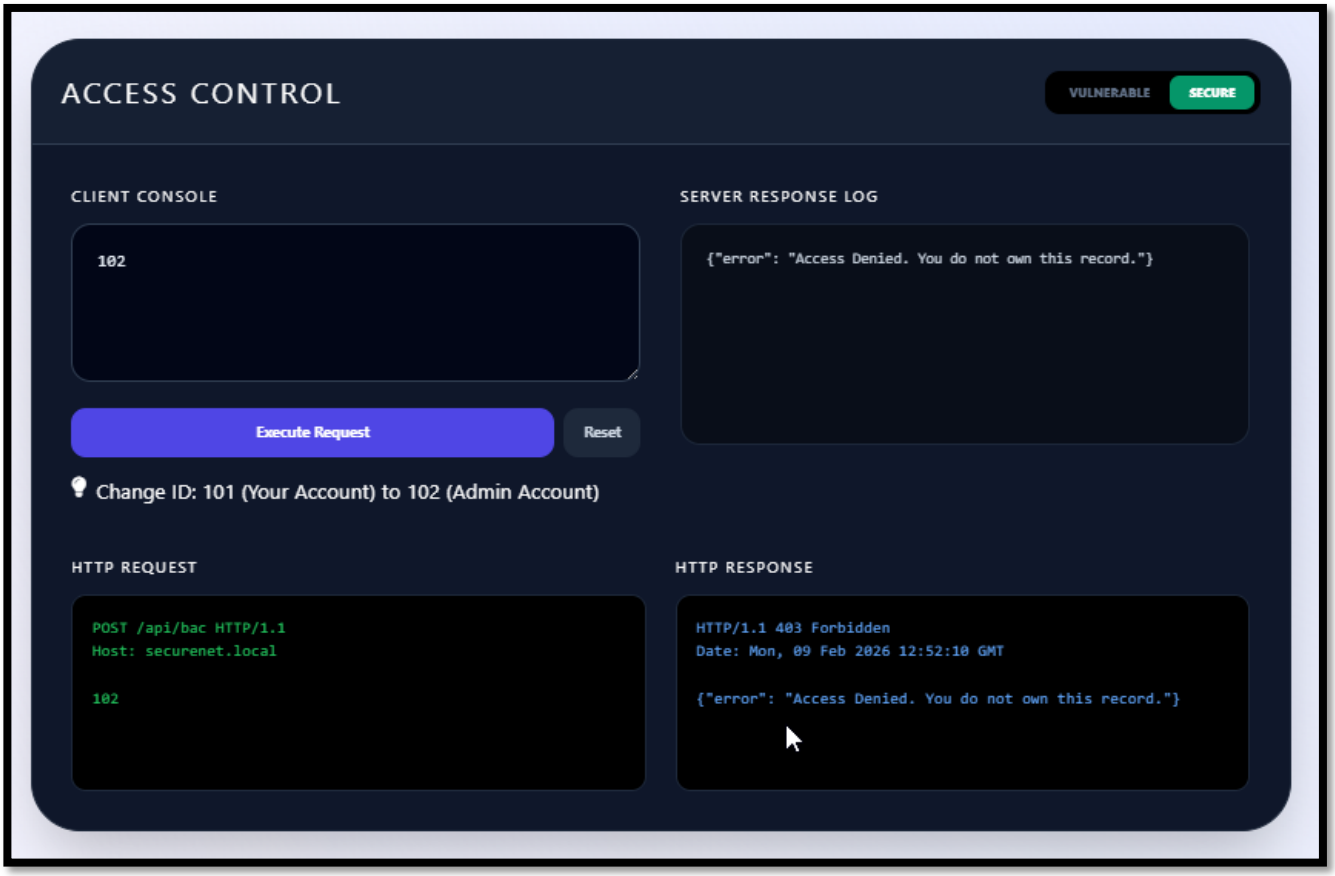
30. **Access Control : In this simulation 101 is the normal user and 102 is admin user , when the vulnerable cod is implemented 101 is able to fetch the data of 102**





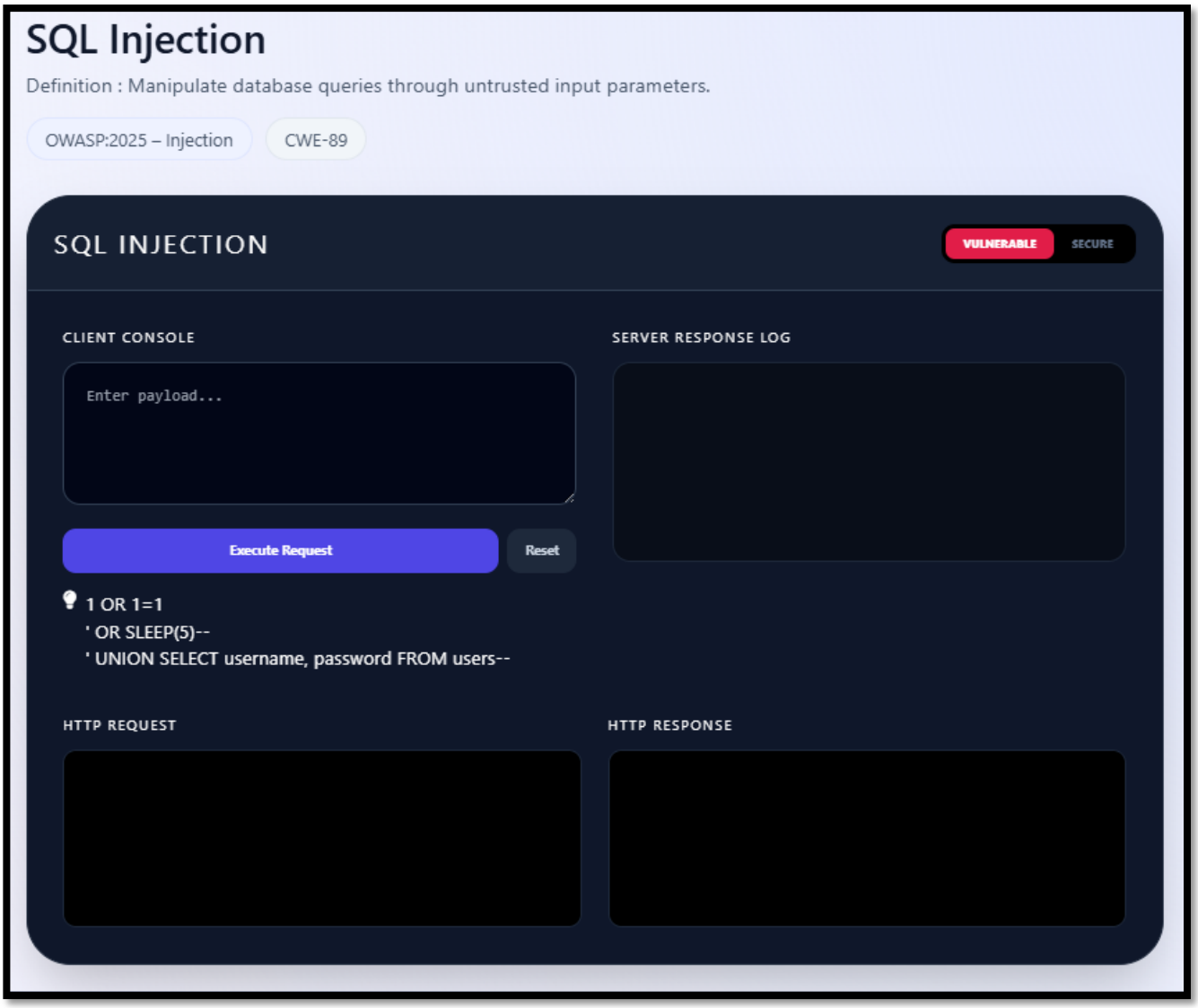
31. **Access Control : When you switch to the secure mode 101 is able to fetch their own data but not of 102.**



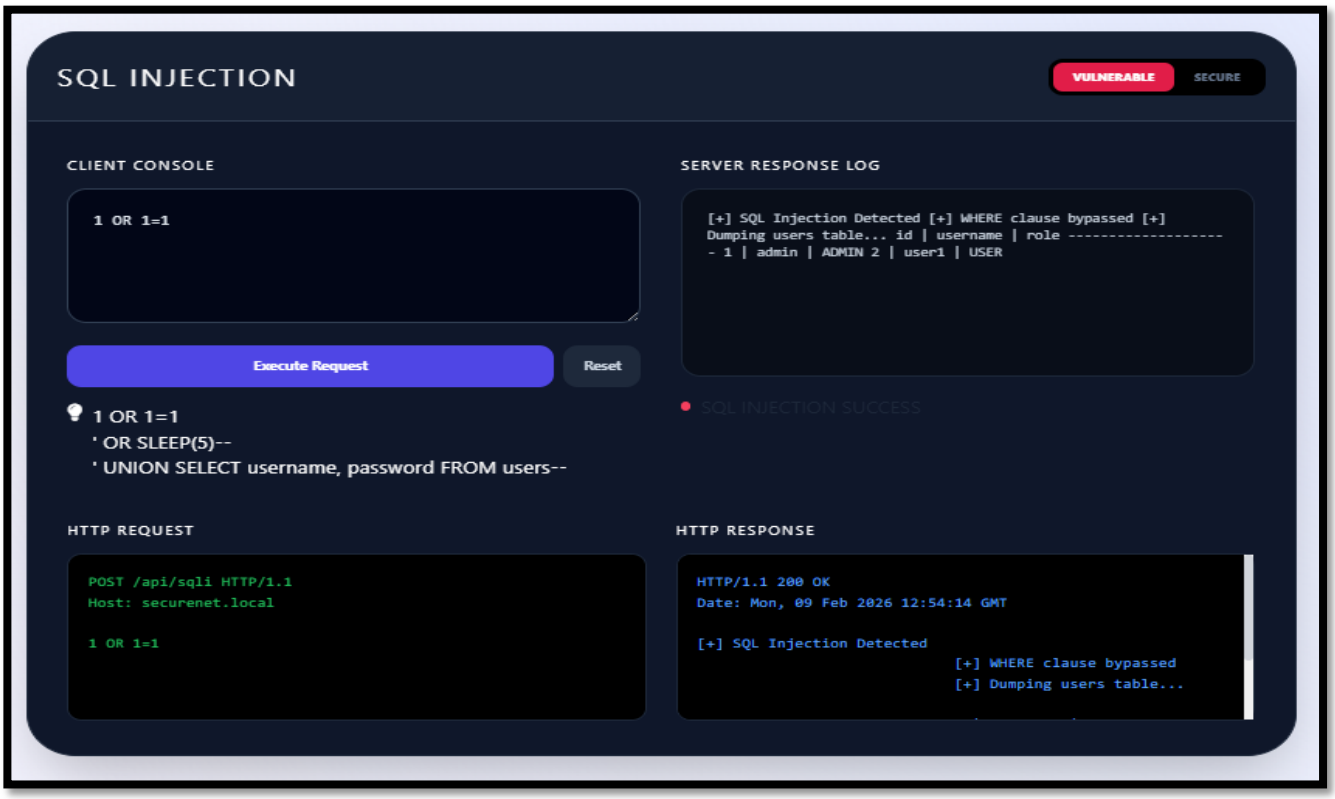


32. **Access Control : Below is the code of secure and vulnerable logic**

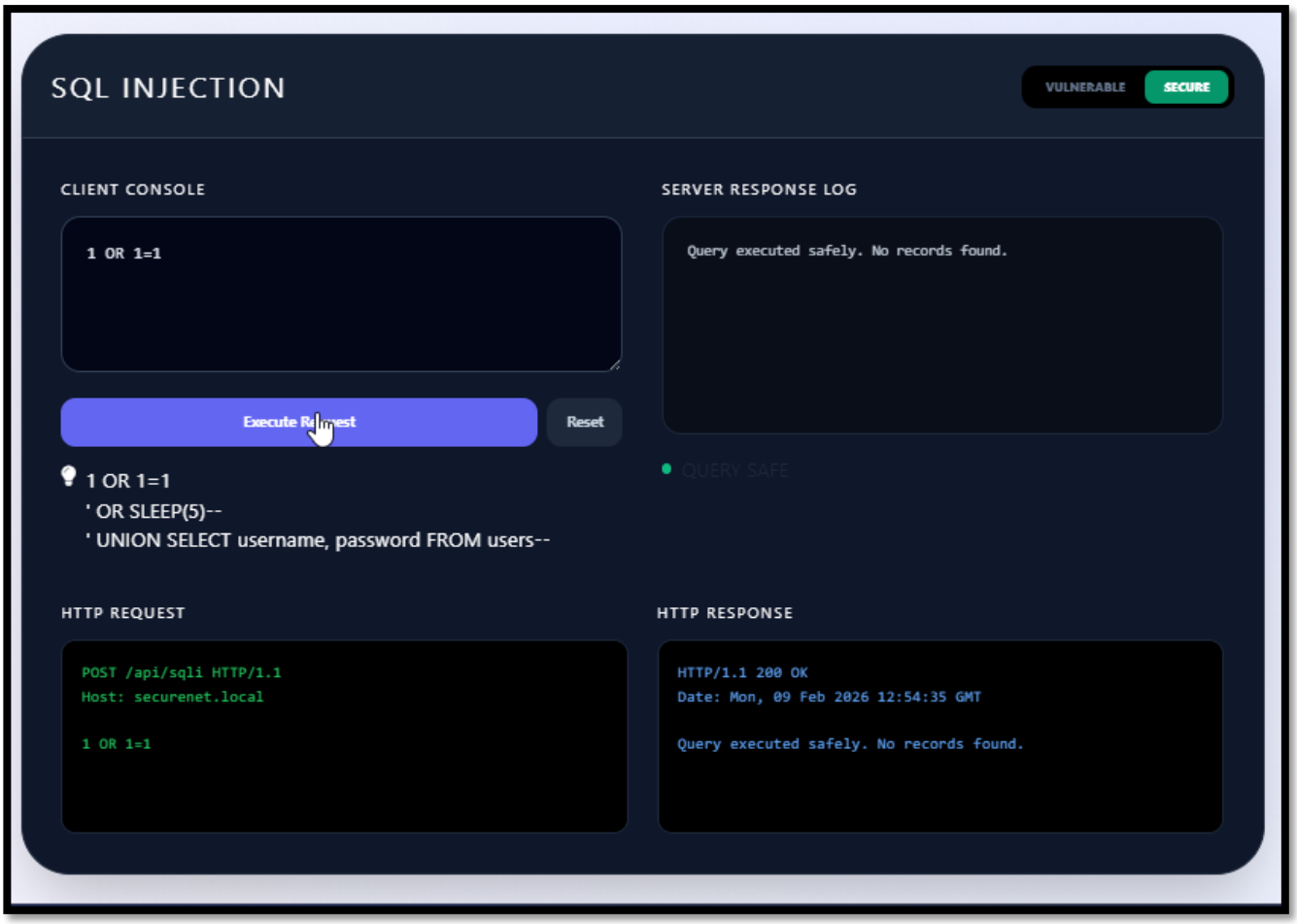
33. **Topic 8 : SQL Injection vulnerability**



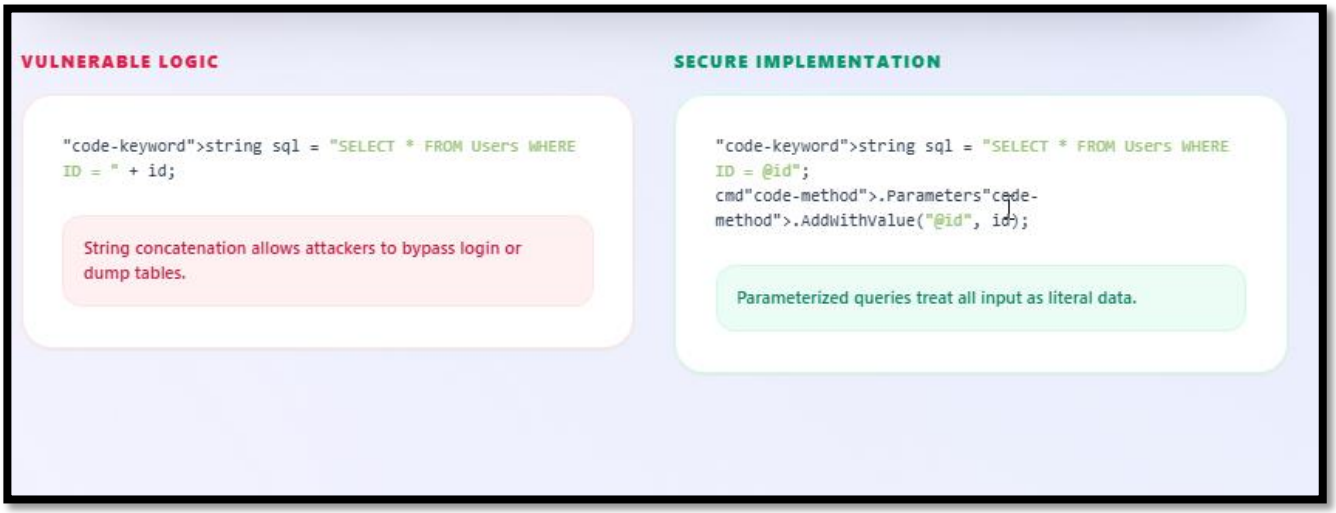
34. **SQL Injection : Enter this payload 1 OR 1=1 in the vulnerable mode**



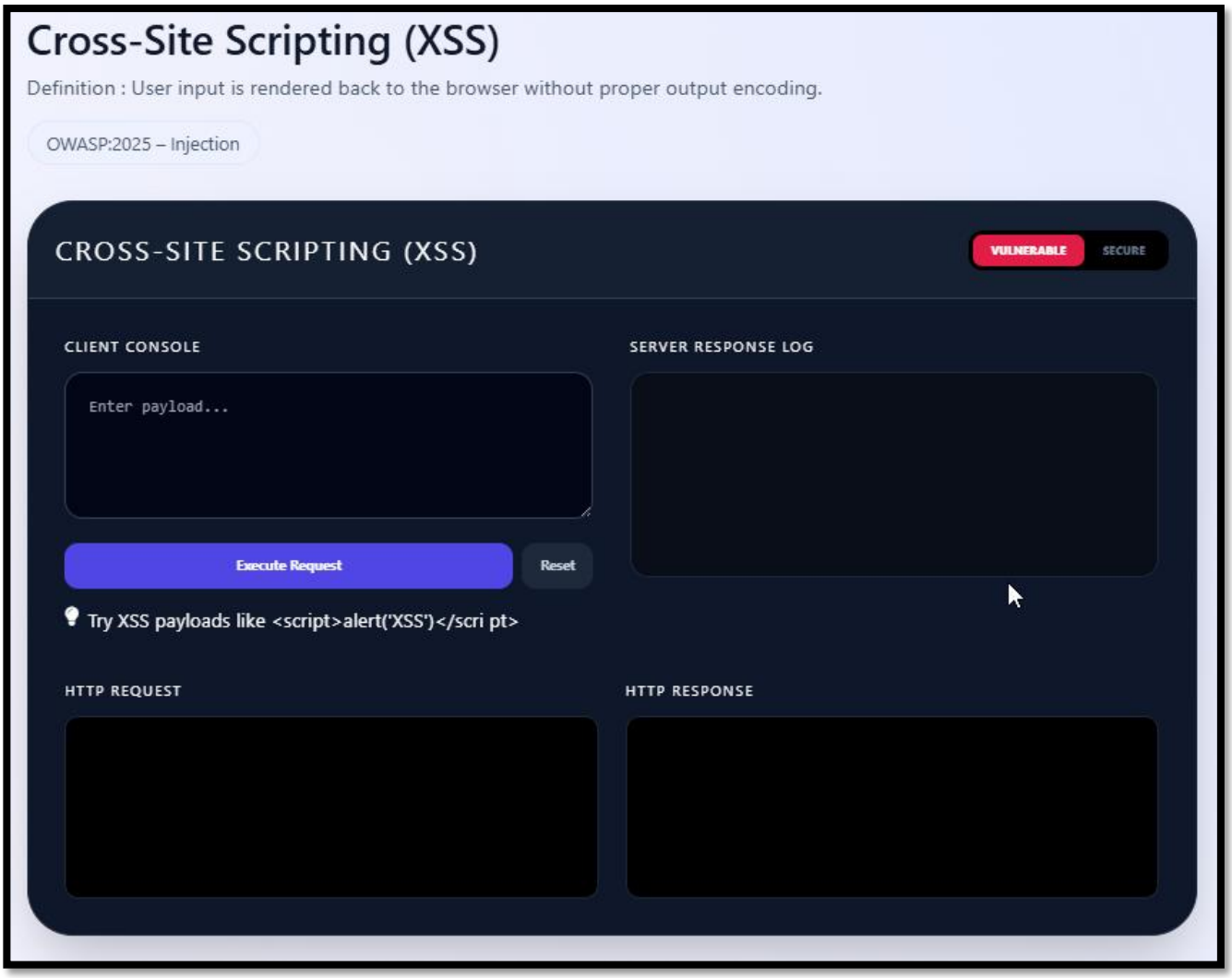
35. **SQL Injection : Switch to the Secure mode and enter the same payload**



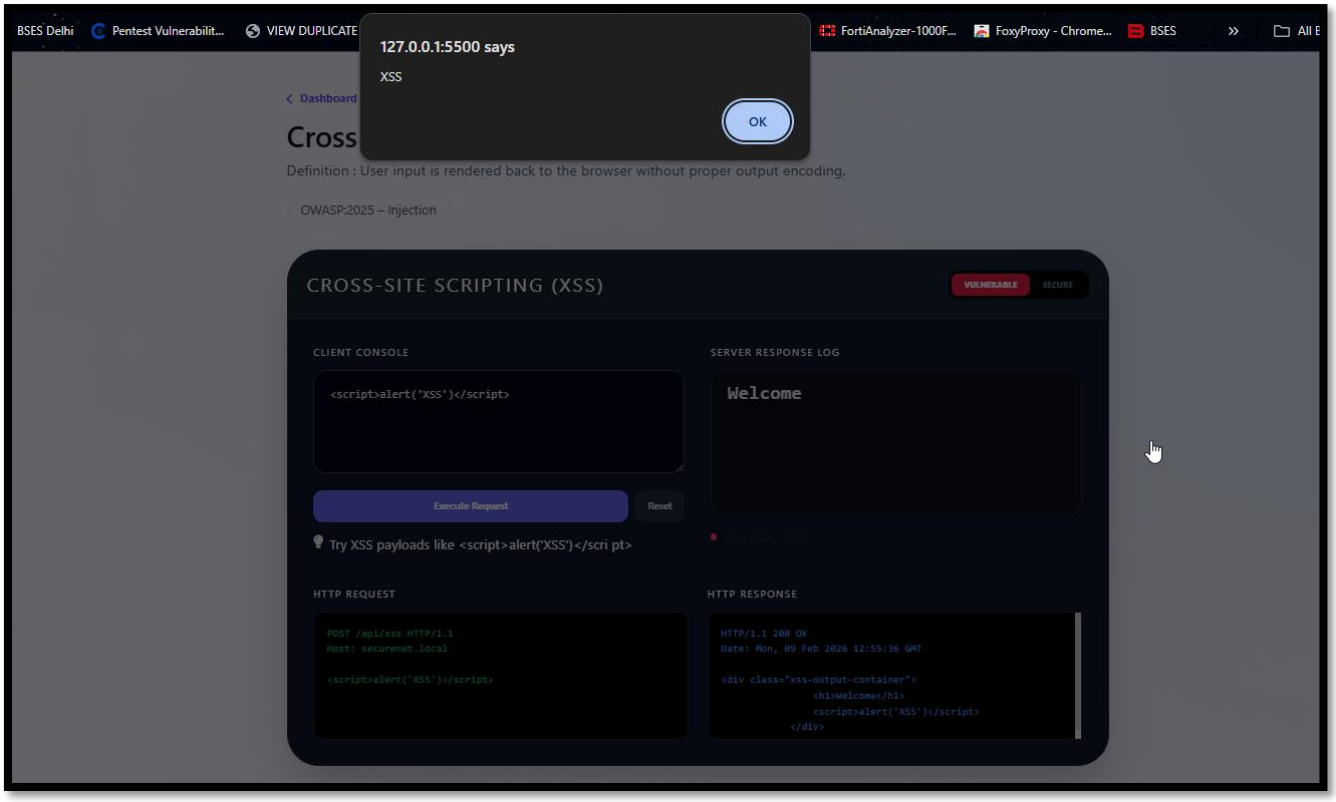
36. **SQL Injection : Below is the secure and vulnerable logic**



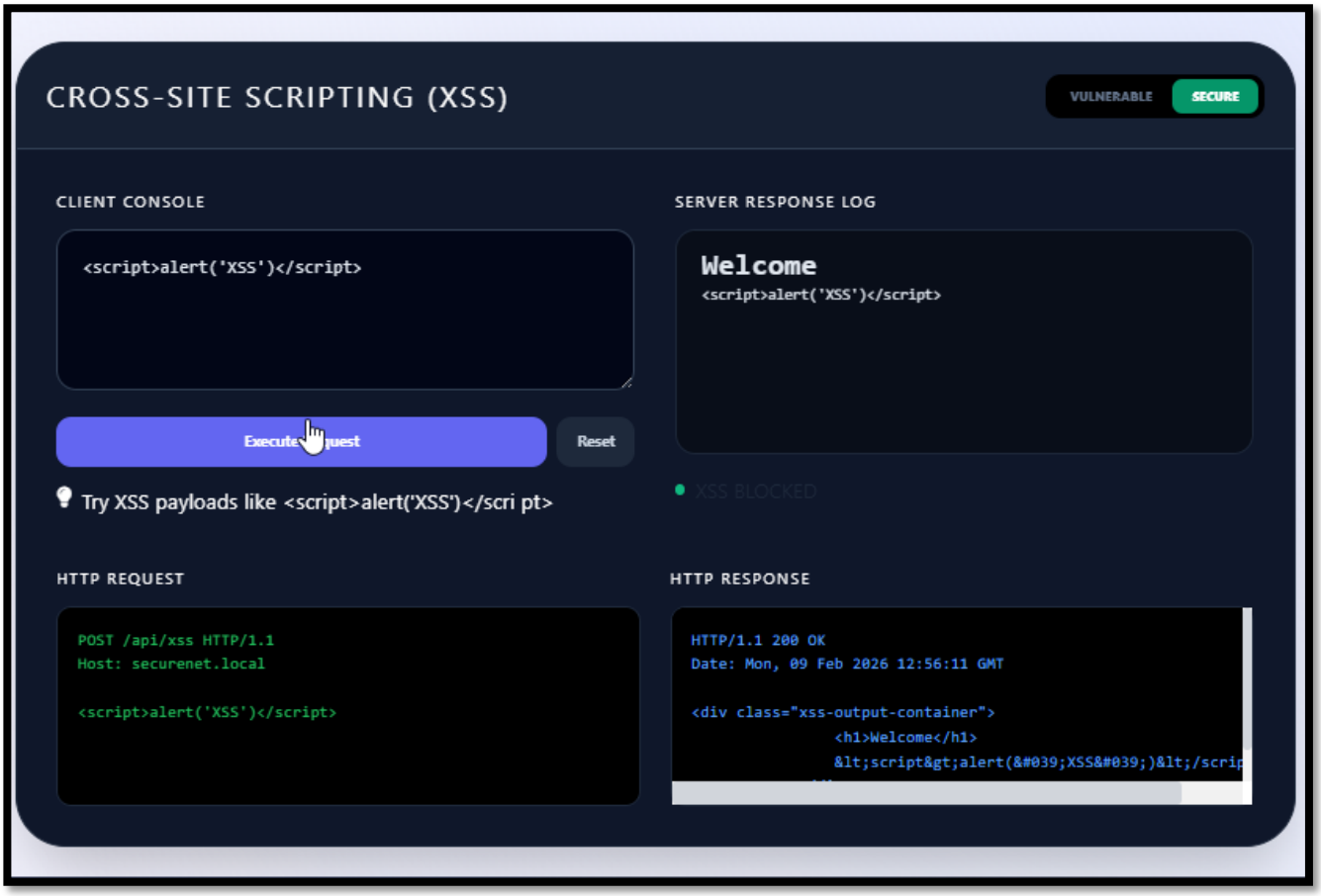
37. **Topic 9 : Cross site scripting (XSS)**



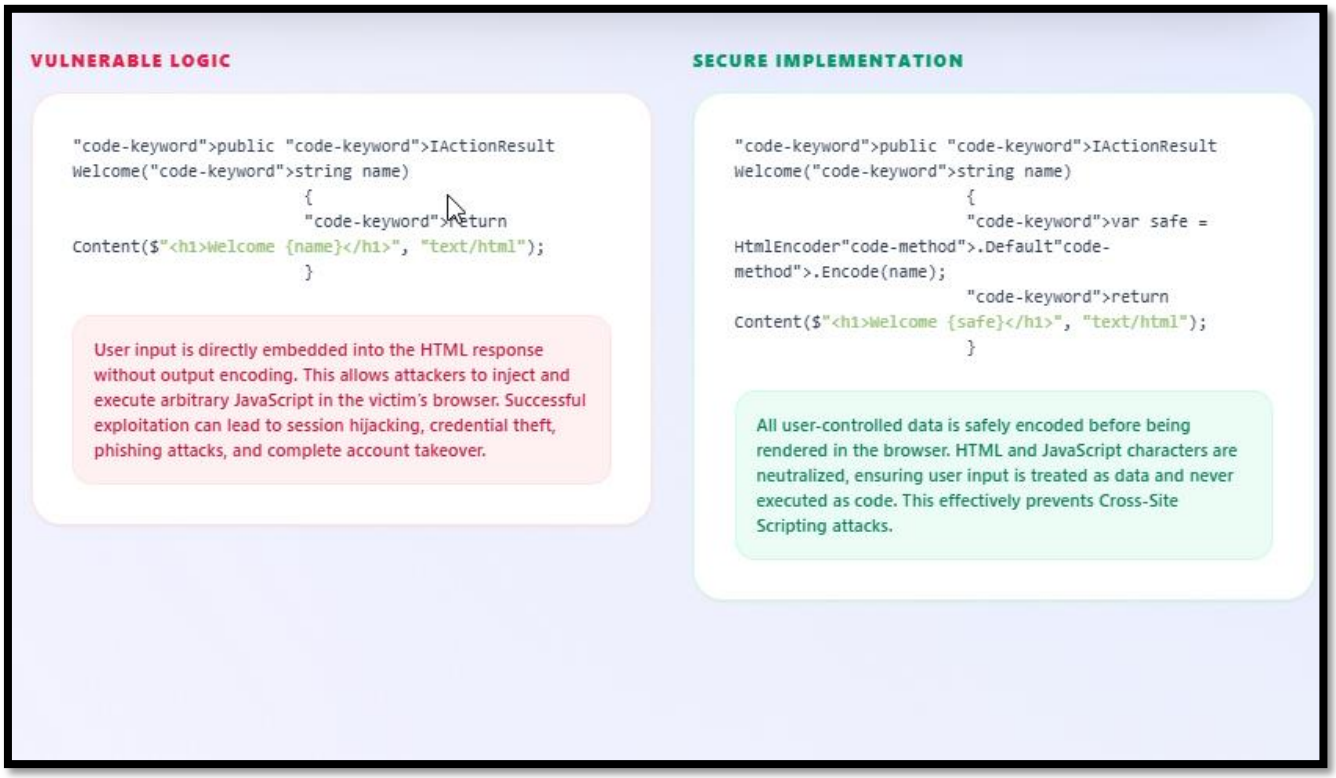
36. **XSS : Enter this payload <script>alert('XSS')</script> in the input section , Vulnerable code execute the payload**



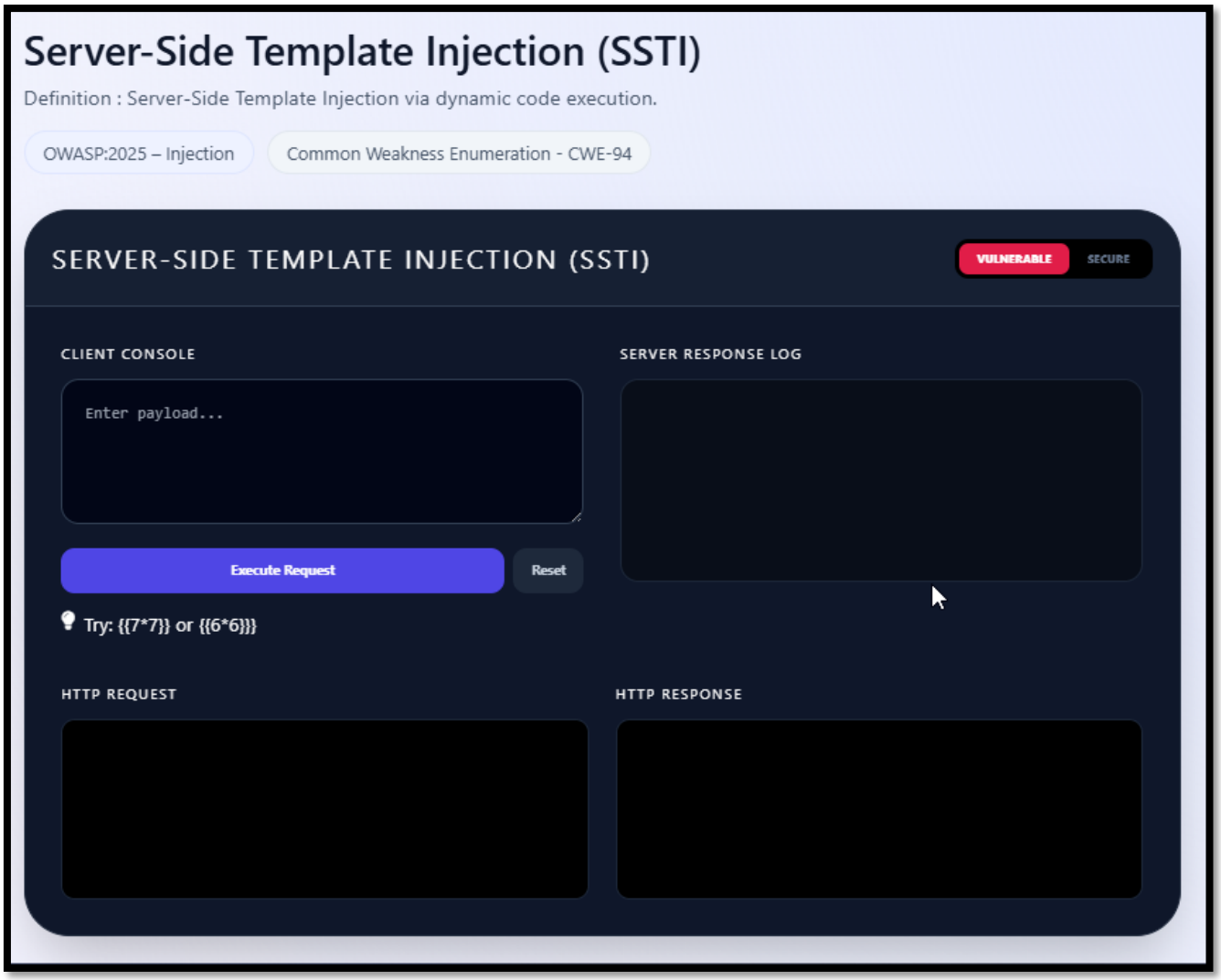
38. **XSS : Enter Same payload in the secure section , secure code implementation does not execute the payload**



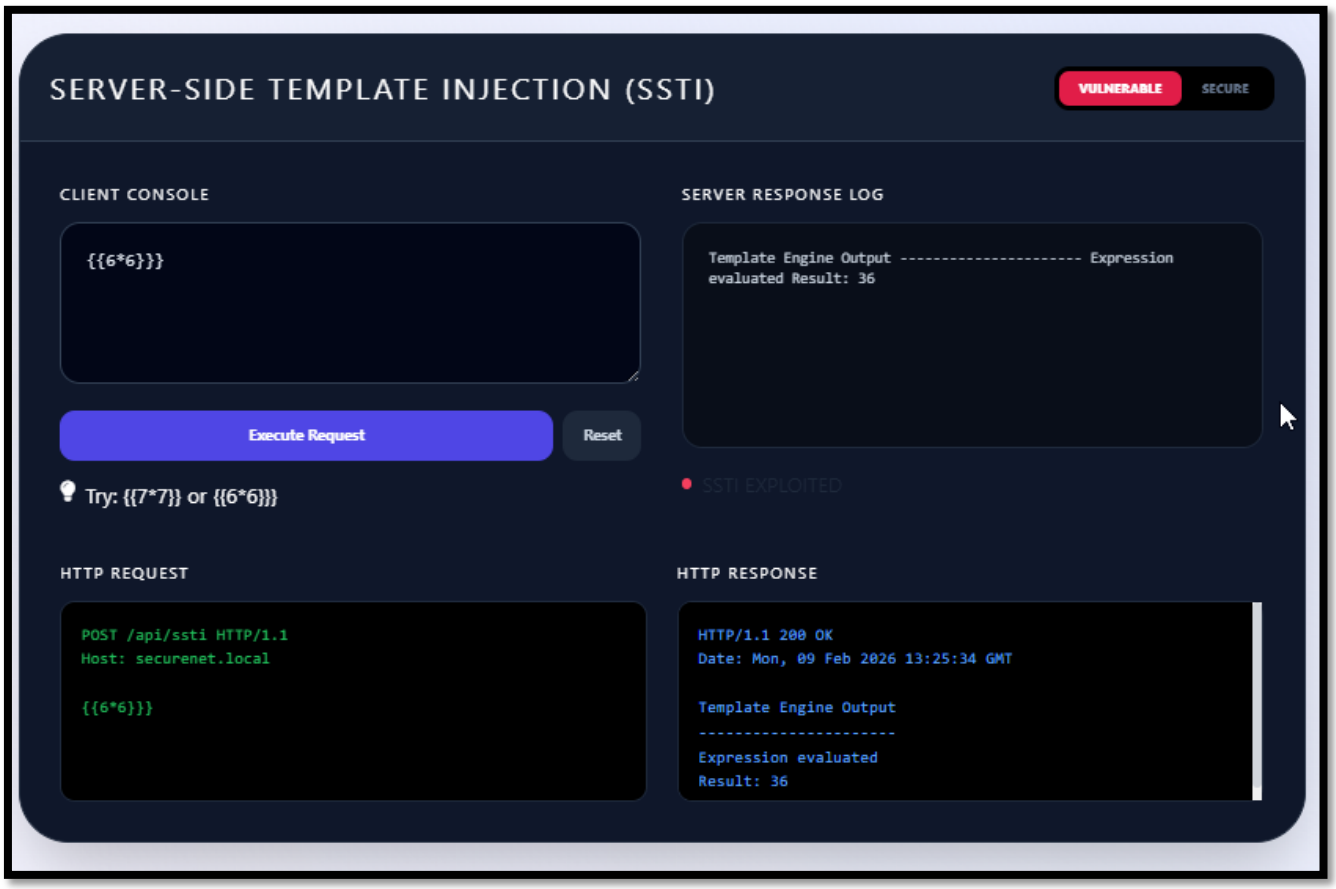
39. **XSS : Below is the secure and vulnerable logic**



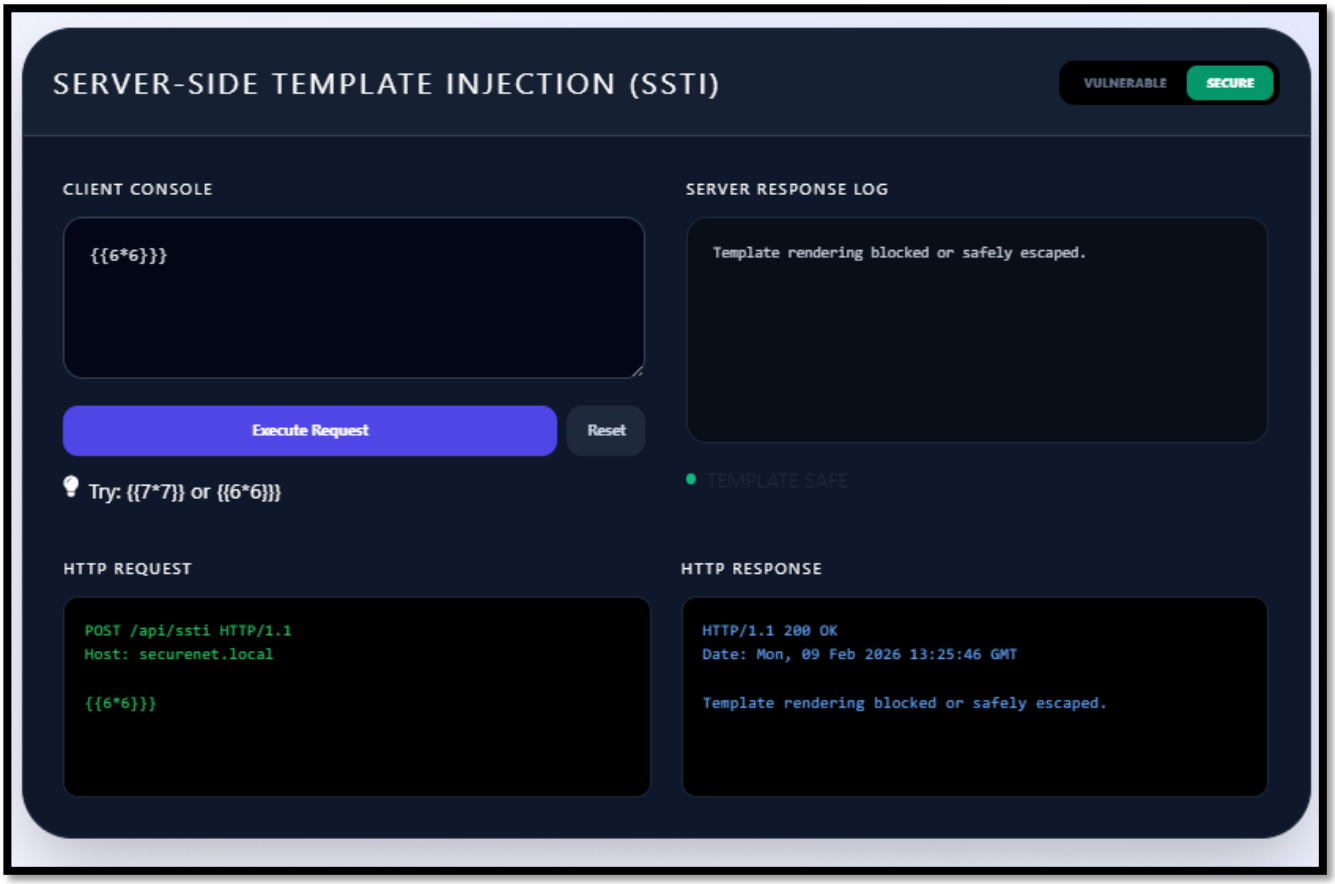
40. **Topic 10 : Server-Side Template Injection (SSTI)**



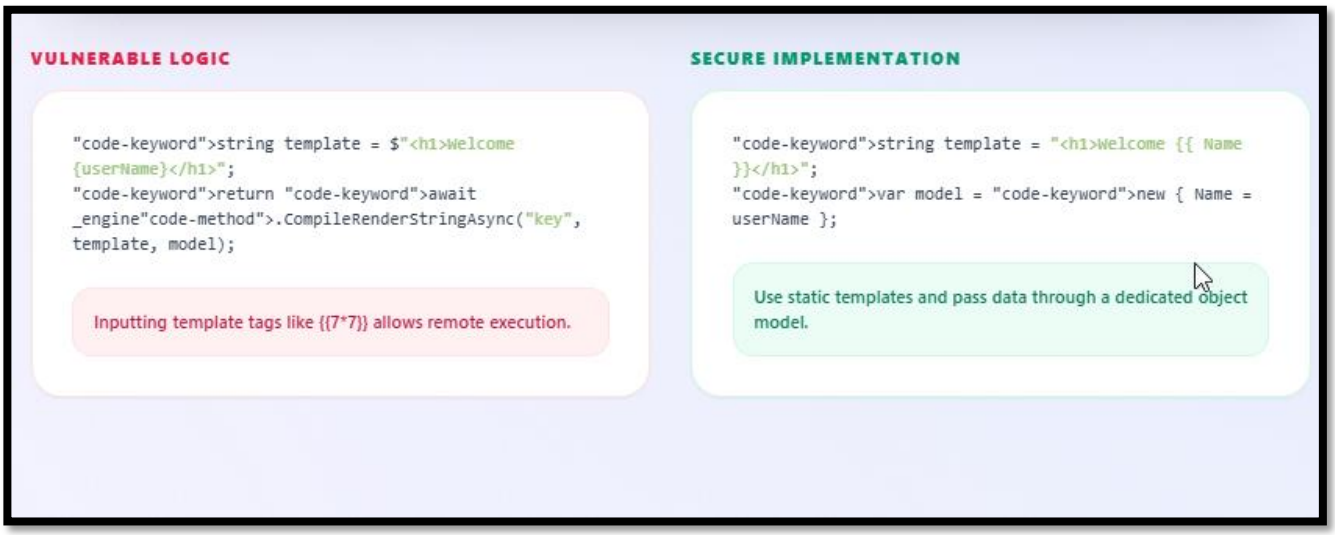
41. **SSTI : Enter this payload {{7*7}} or {{6*6}} in the input section , Vulnerable code executes the SSTI payload**



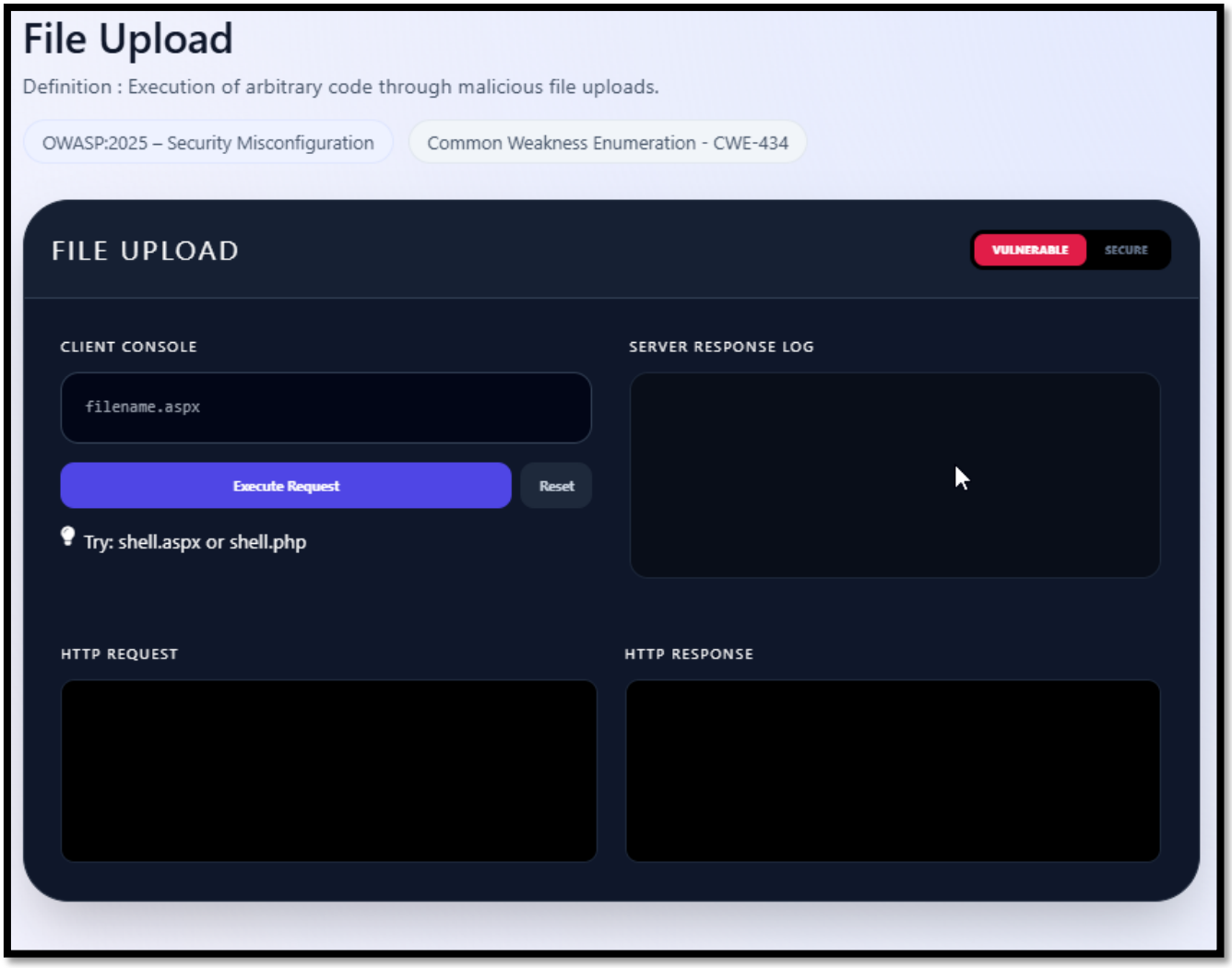
42. SSTI :Switch to the secure mode and enter the same payload , Secure code does not executes the SSTI payload



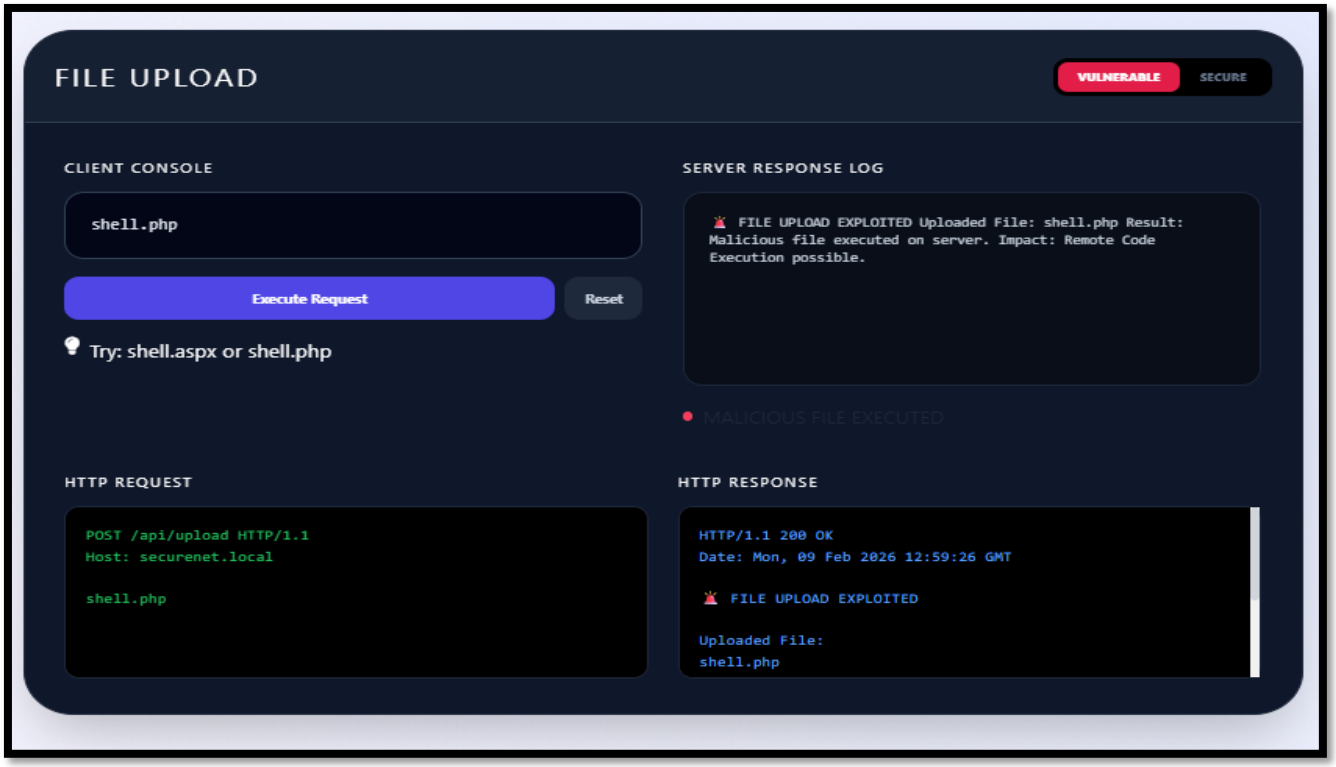
43 . SSTI : Below is the secure and vulnerable logic



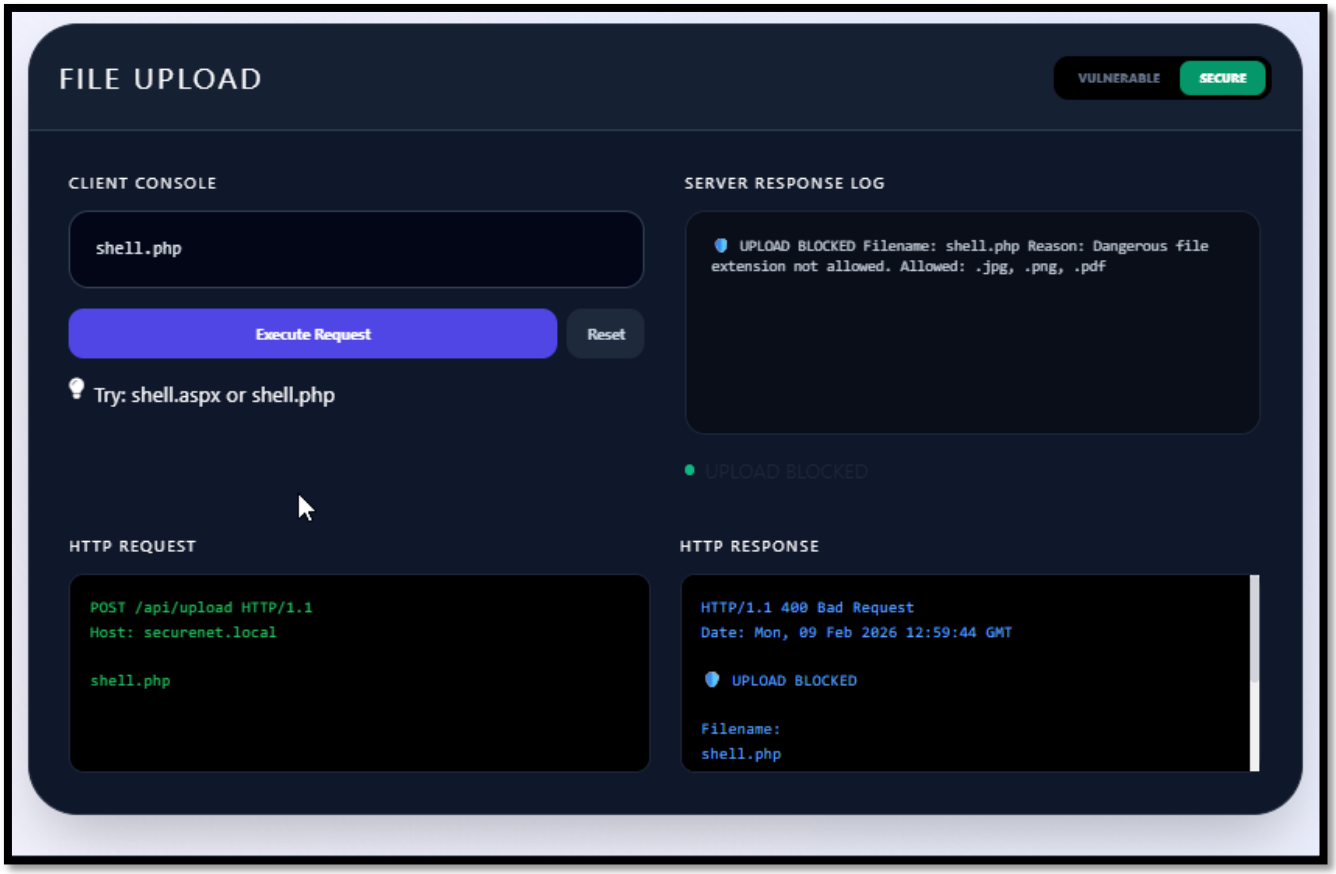
44. Topic 11 :File Upload



45.File Upload : Enter this payload shell.php or shell.aspx in the input section , Vulnerable code accepts the .php or .aspx

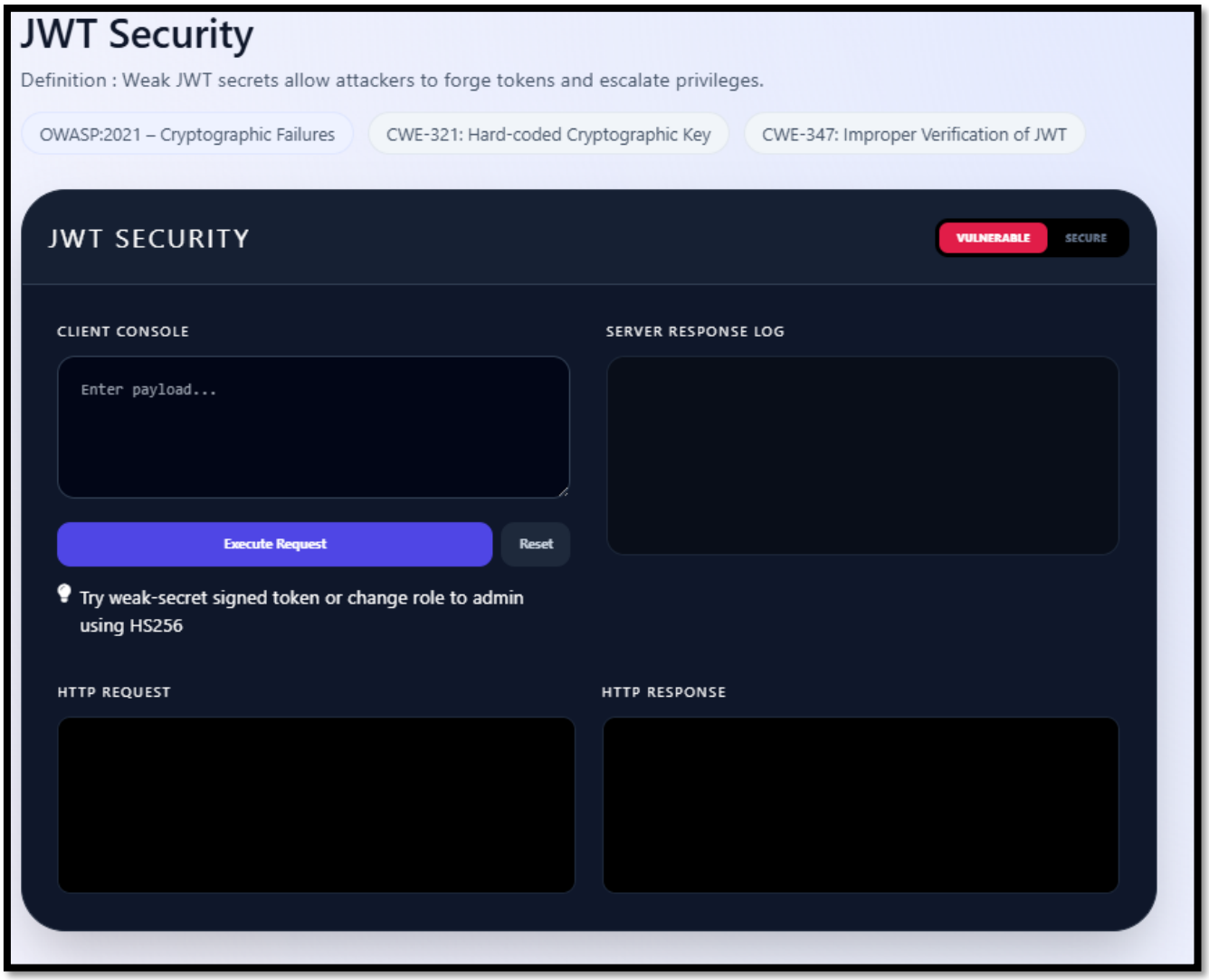


46.File Upload : Switch to the secure mode and enter the same payload , Secure code does not accepts the .php and .aspx

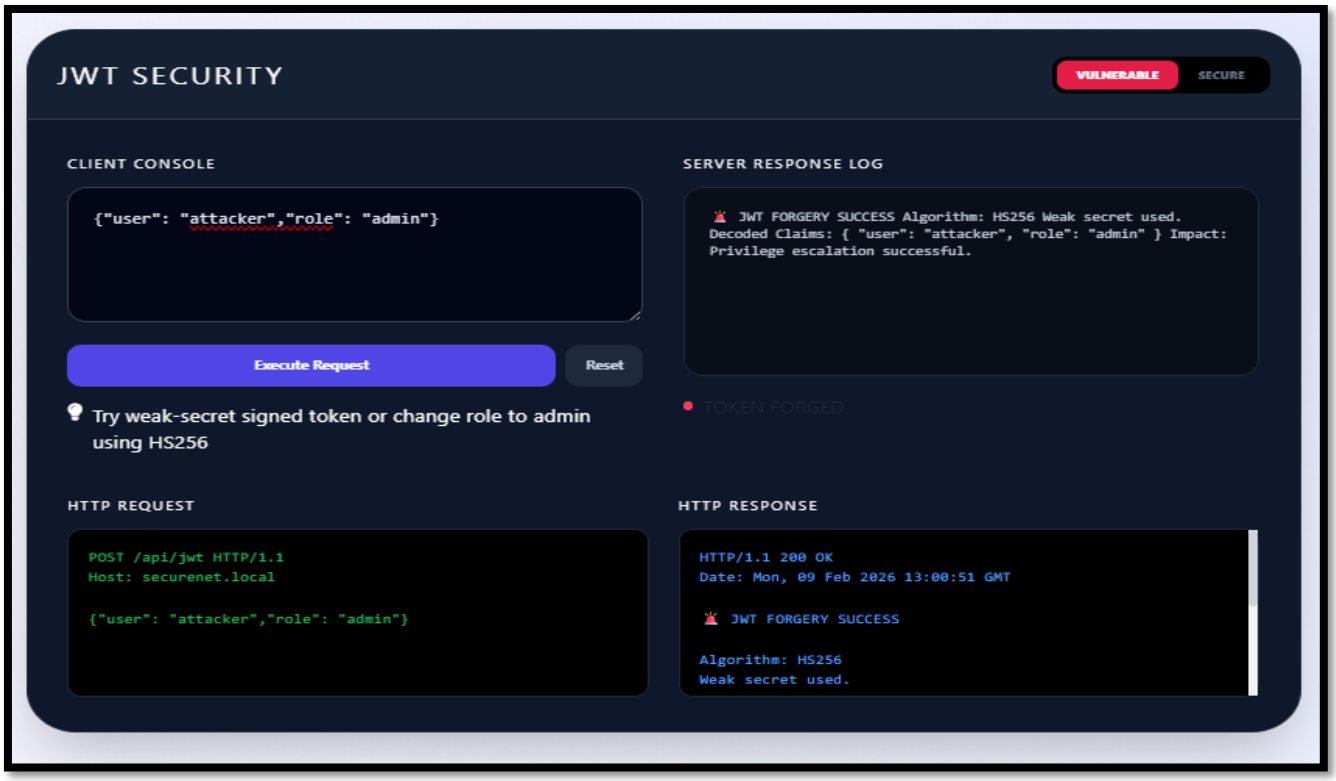


47. File Upload : Below is the secure and vulnerable logic

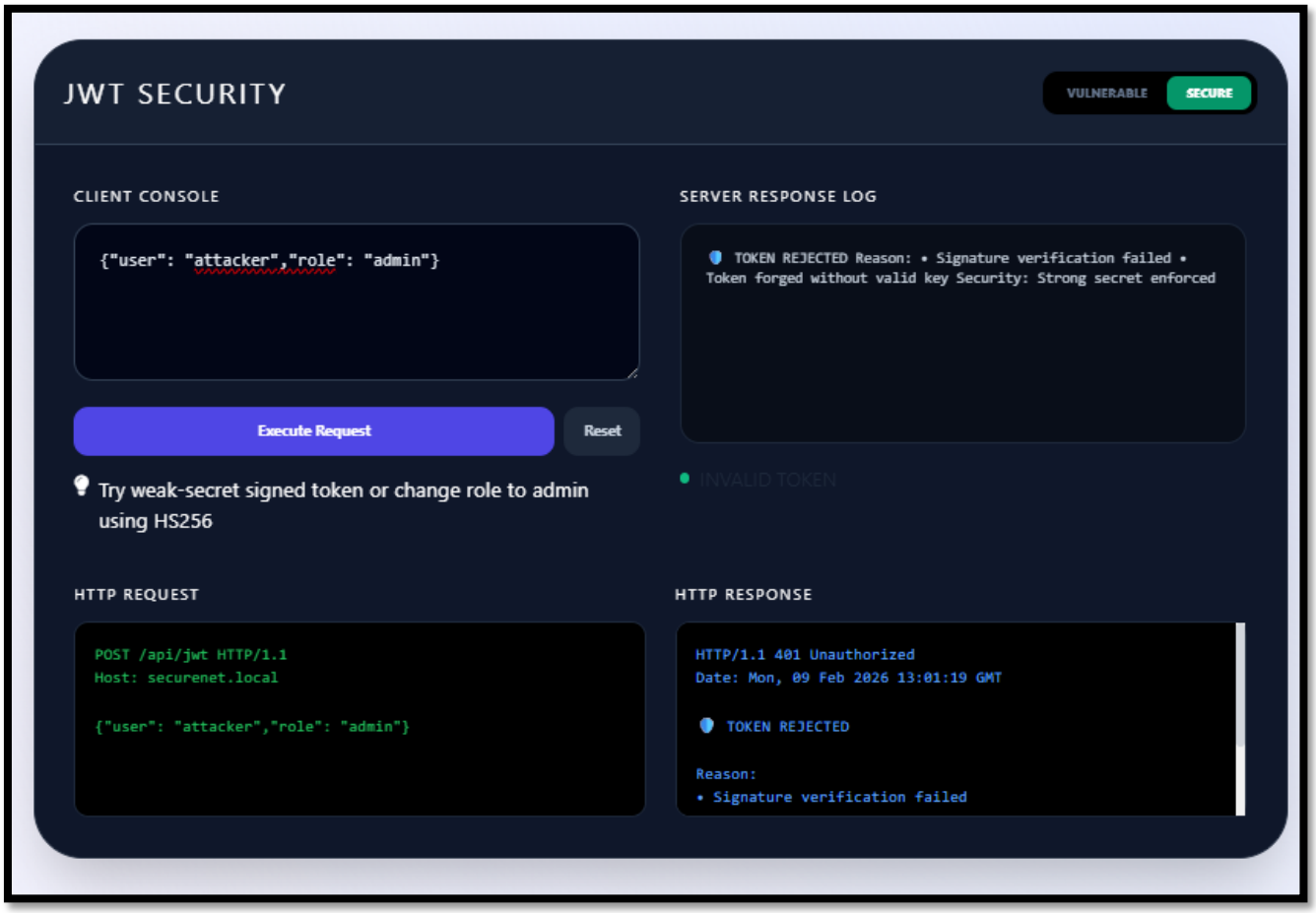
48. **Topic 13 : JWT Security**



49. **JWT Security : Enter this payload in the input section {"user": "attacker","role": "admin"}**

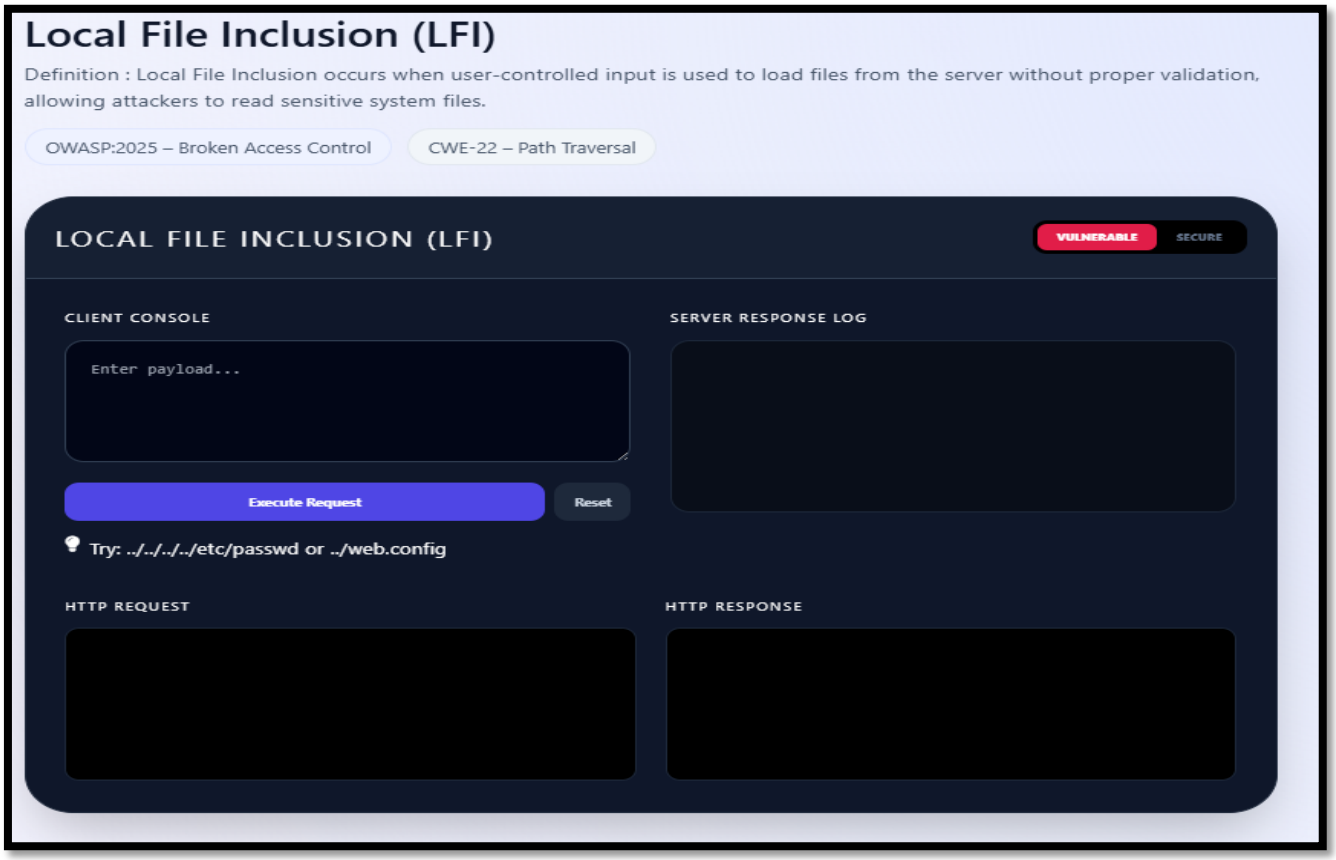


50. JWT Security : Now switch to the secure mode and enter the same payload

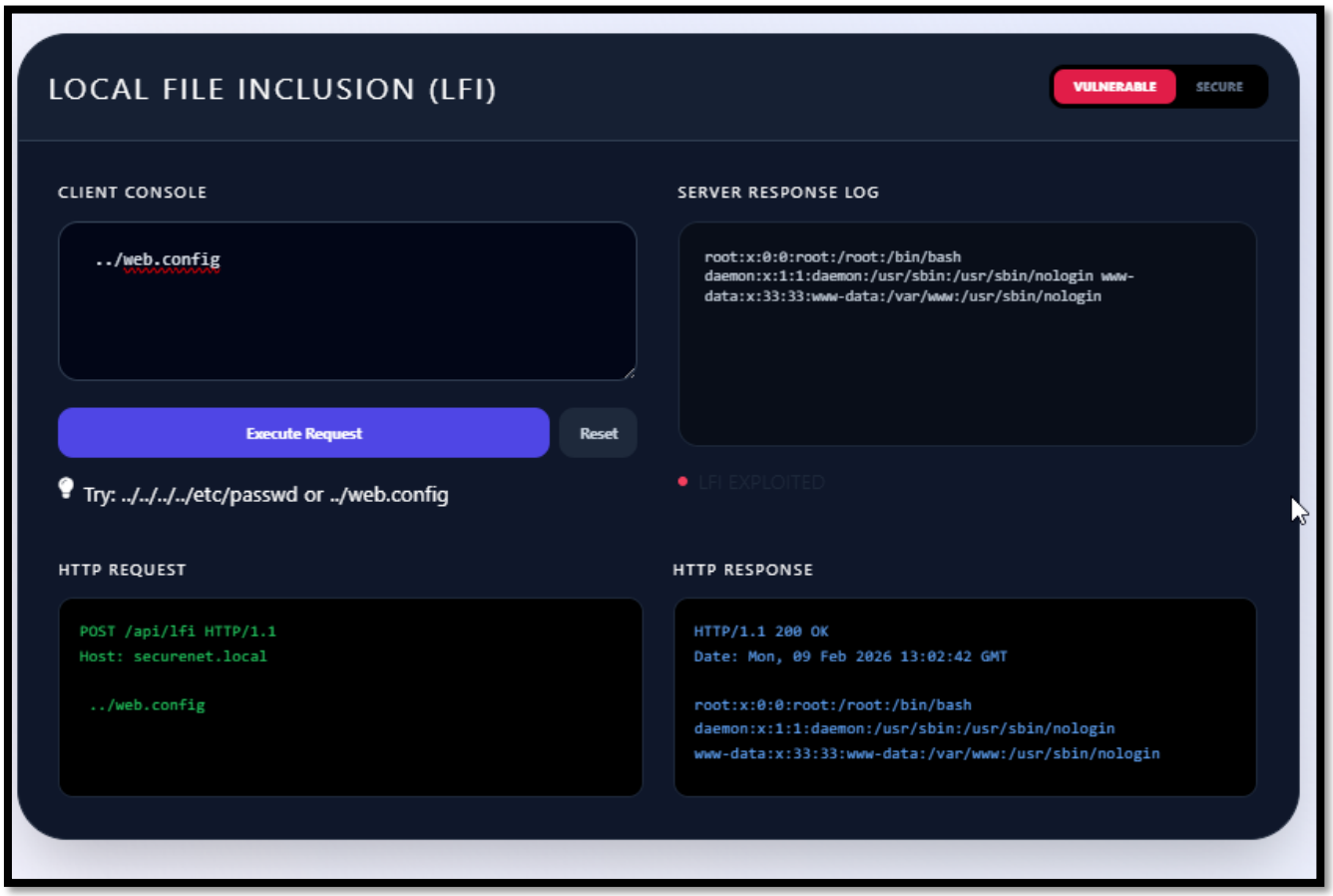


51. JWT Security : Below is the secure and vulnerable logic

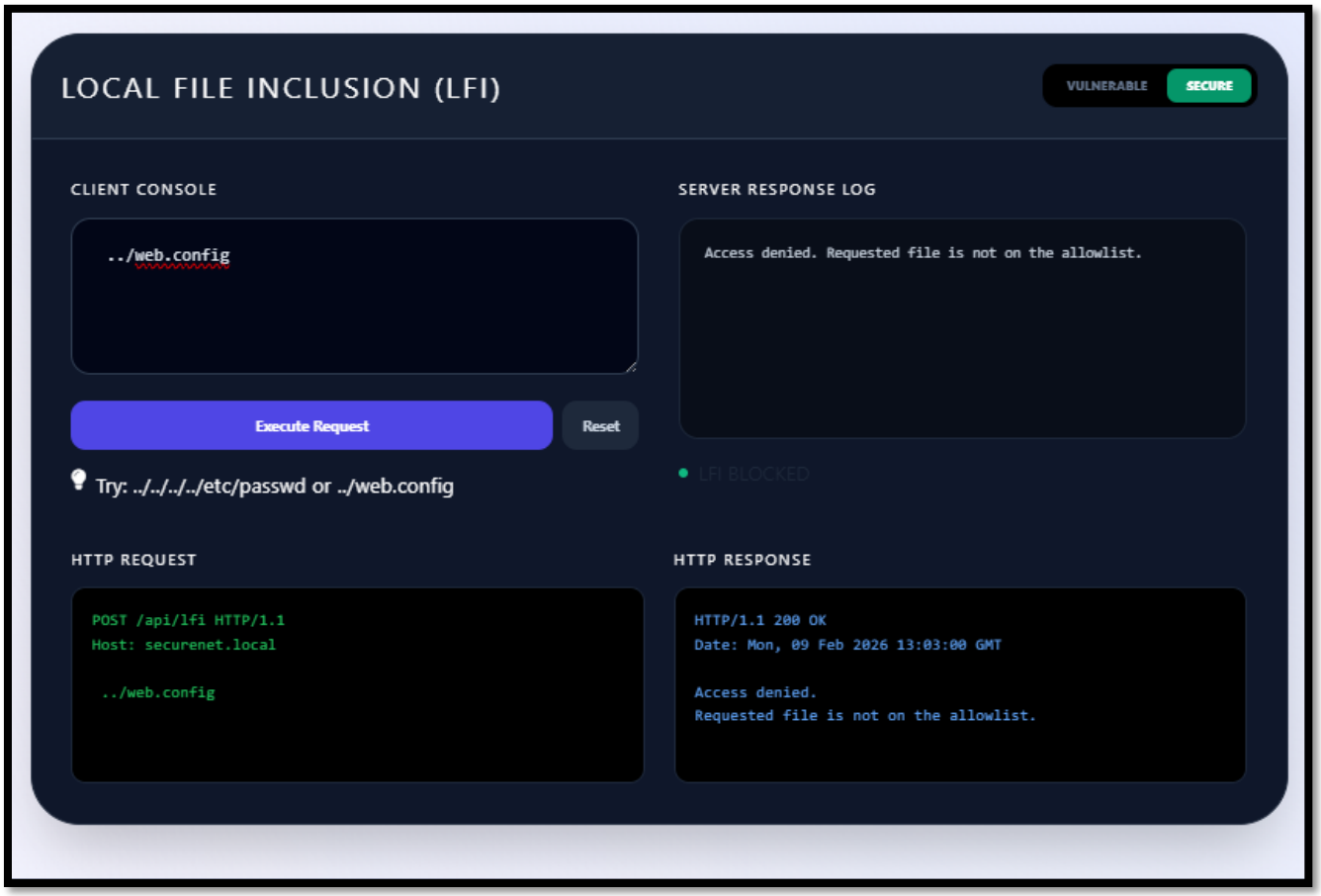
52. Topic 13 :Local file inclusion Vulnerability (LFI)



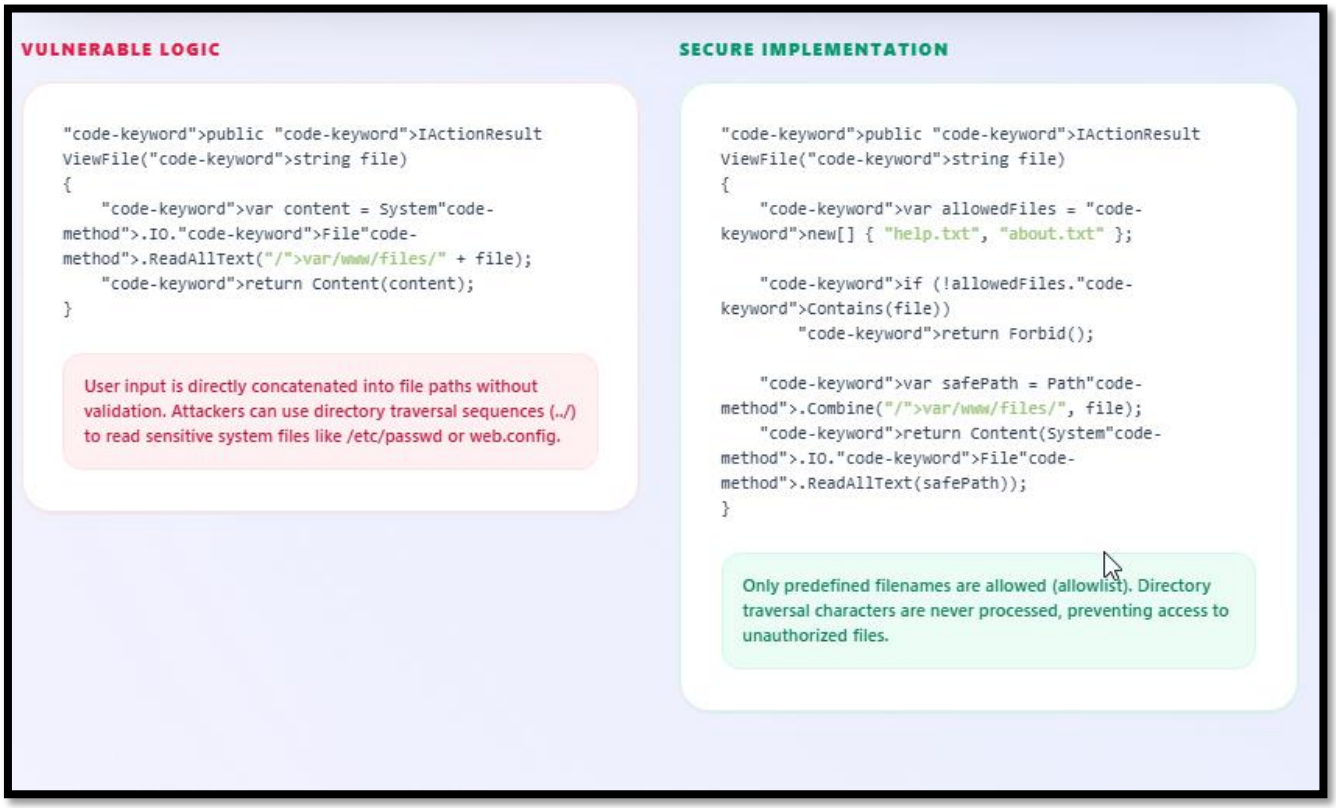
53. LFI : Enter this payload in the input section ../../../../etc/passwd or ../web.config



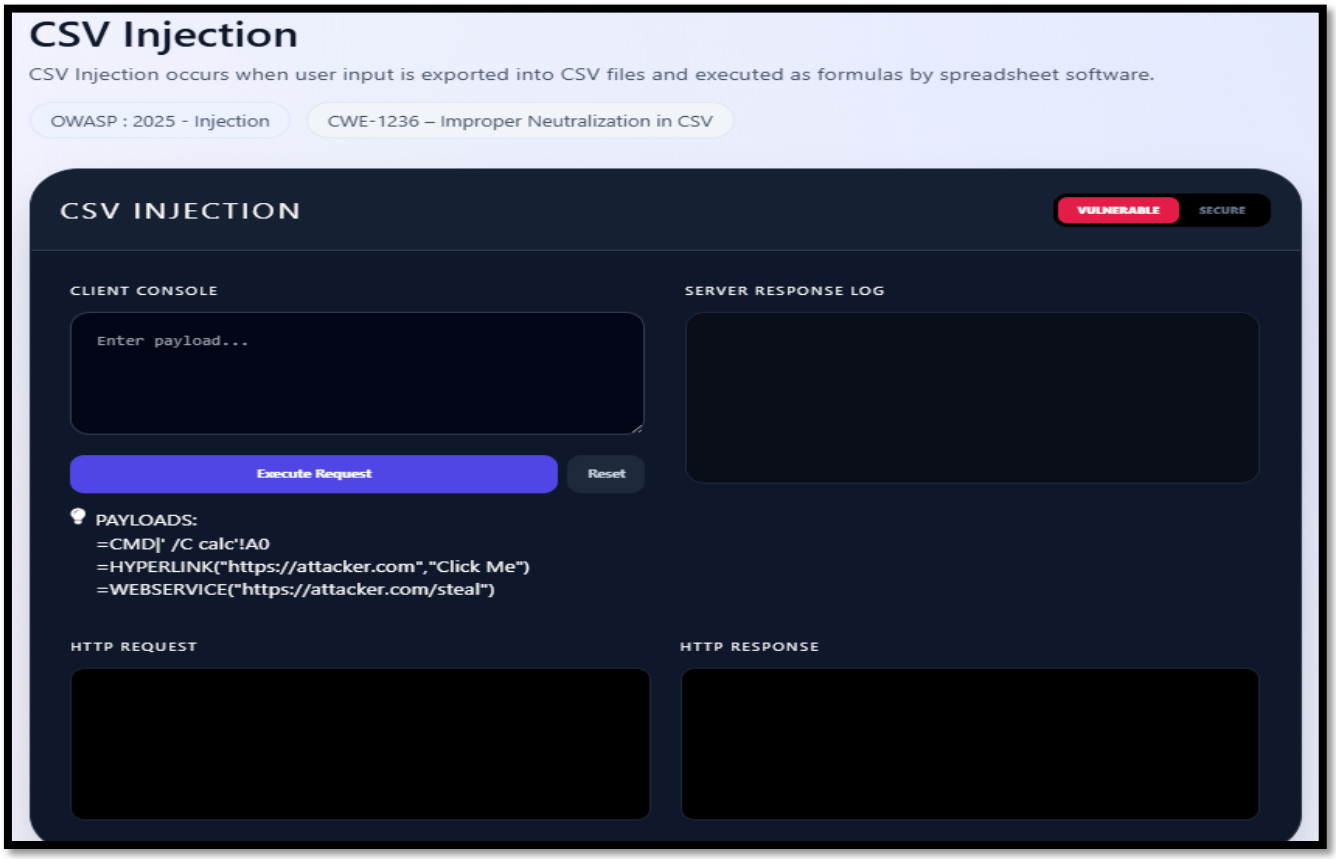
54. LFI :Now switch to the Secure mode and enter the same payload



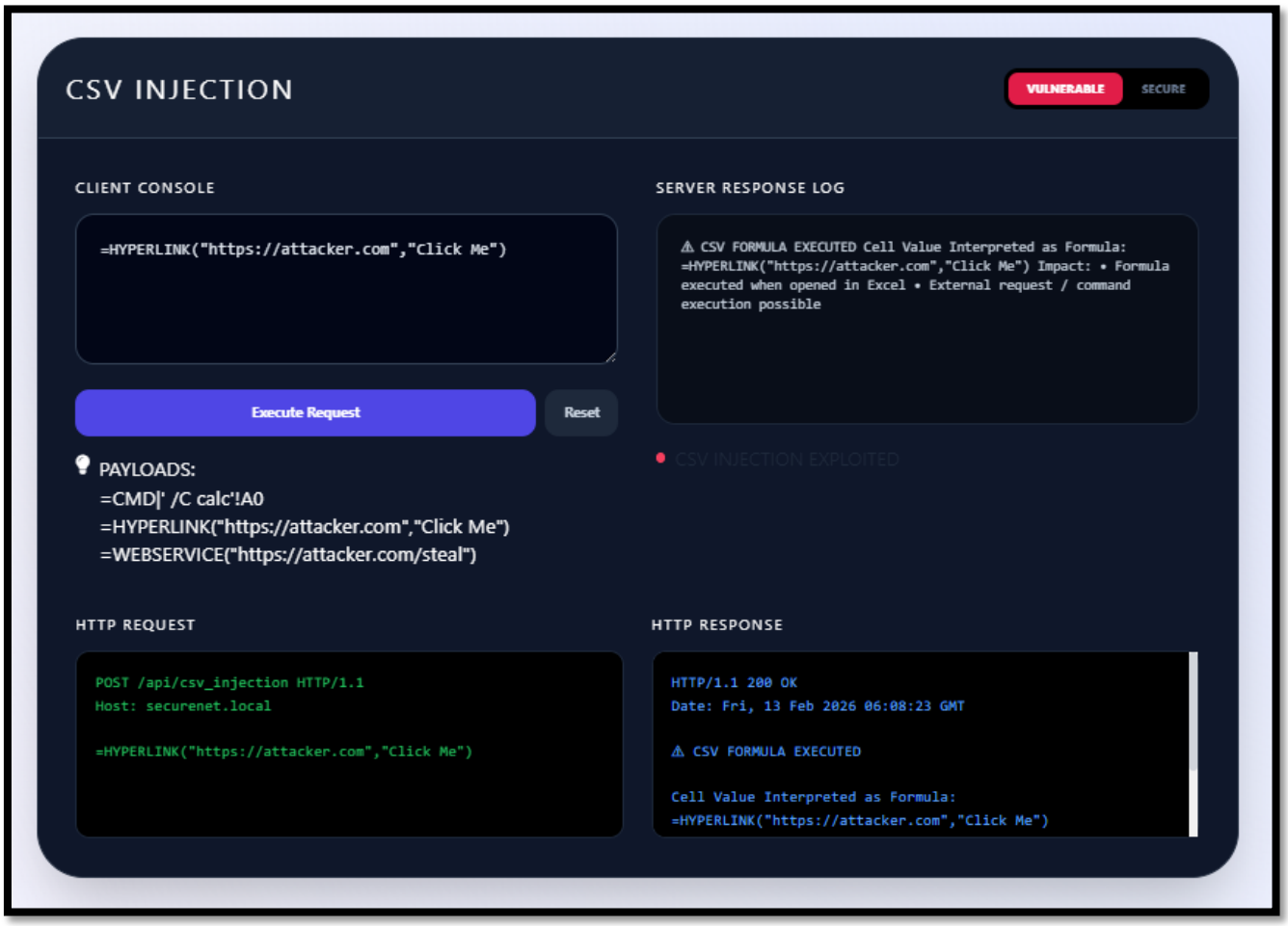
55.LFI : Below is the secure and vulnerable logic



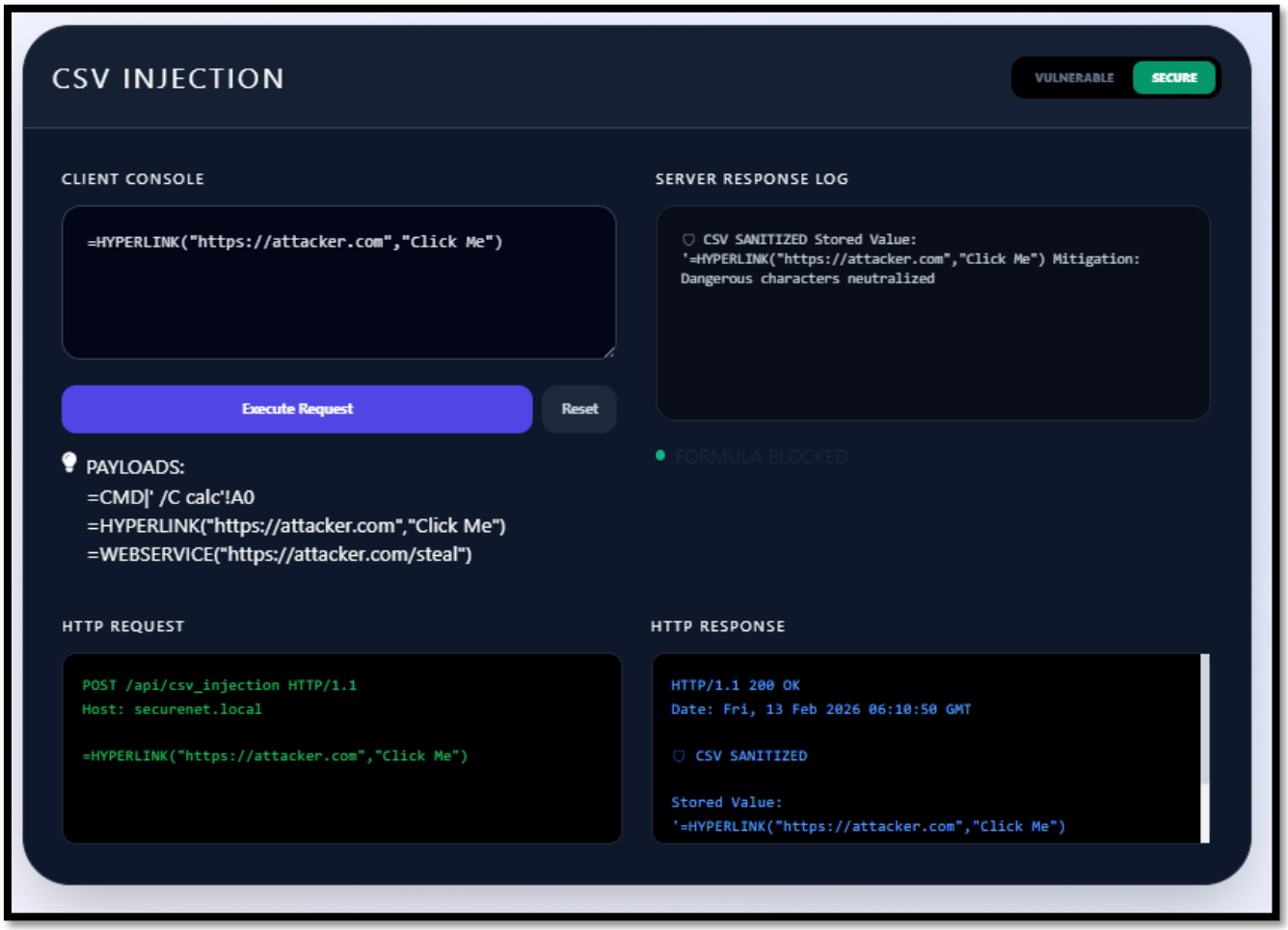
56. Topic 14 : CSV Injection



57. CSV Injection : Enter this payload =HYPERLINK("https://attacker.com","Click Me")



58. CSV Injection : Now Switch to Secure mode and enter the same payload



59. CSV Injection : Below is the secure and vulnerable logic

Thank you