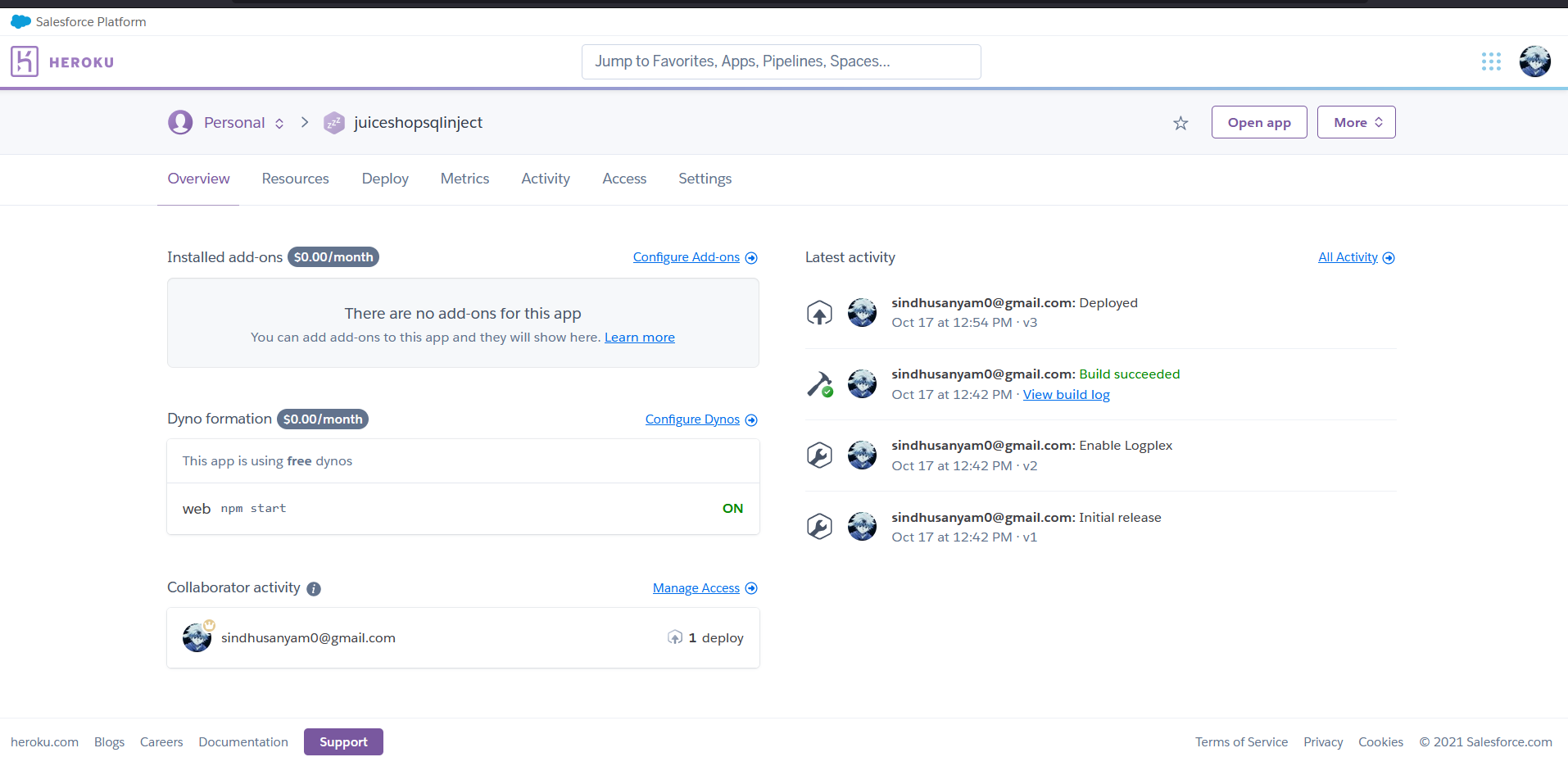
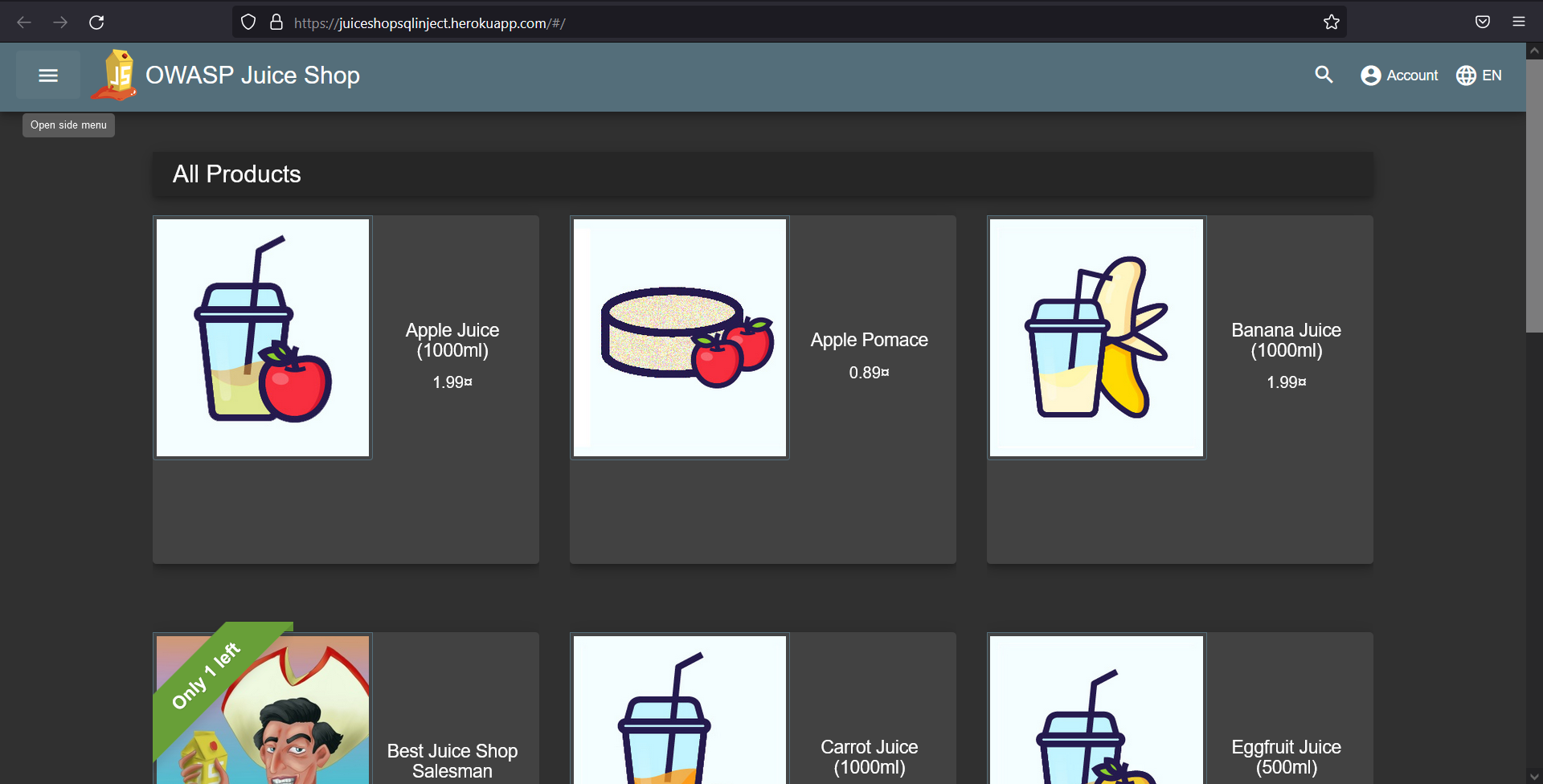
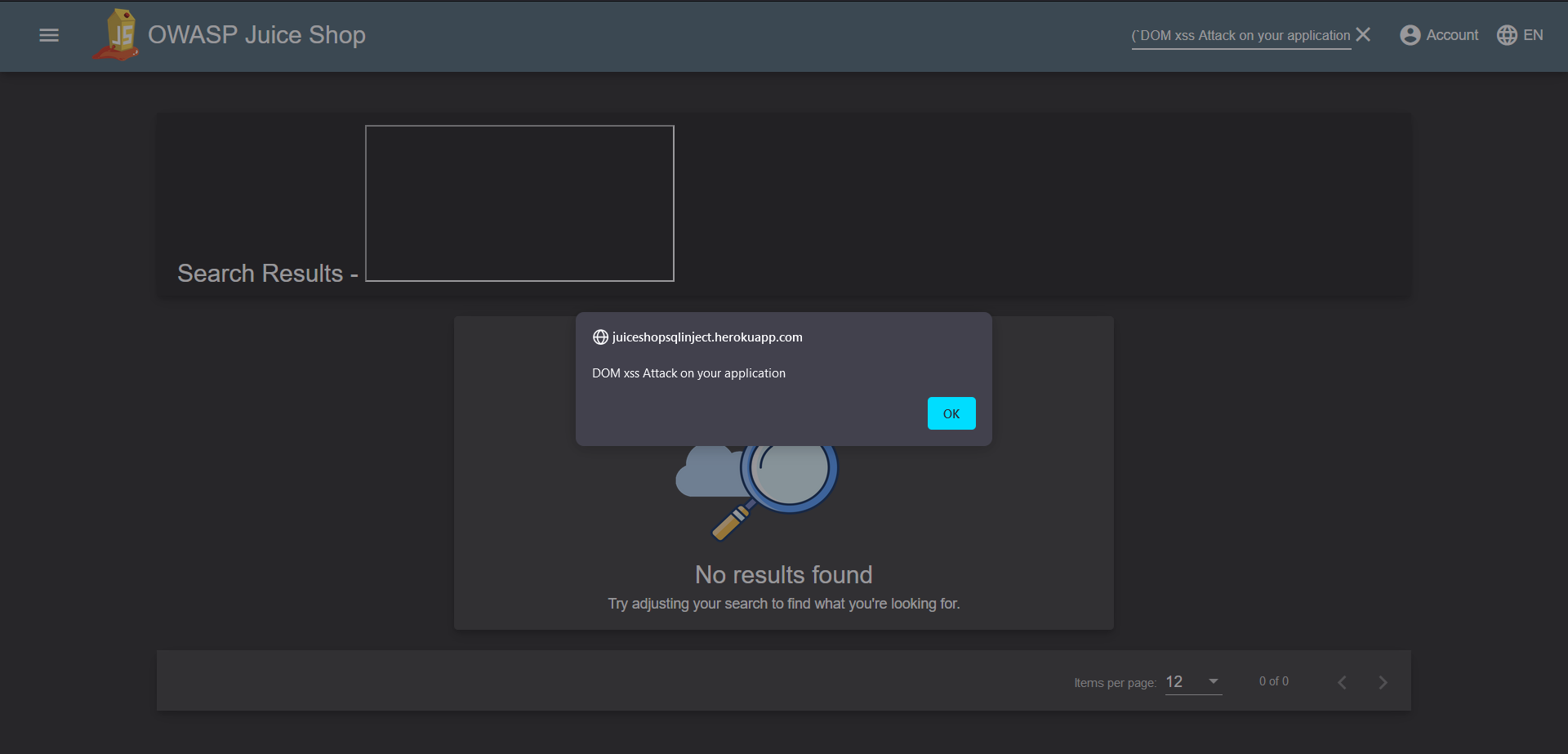
**Deploying the OWASP Juice Shop with Heroku**





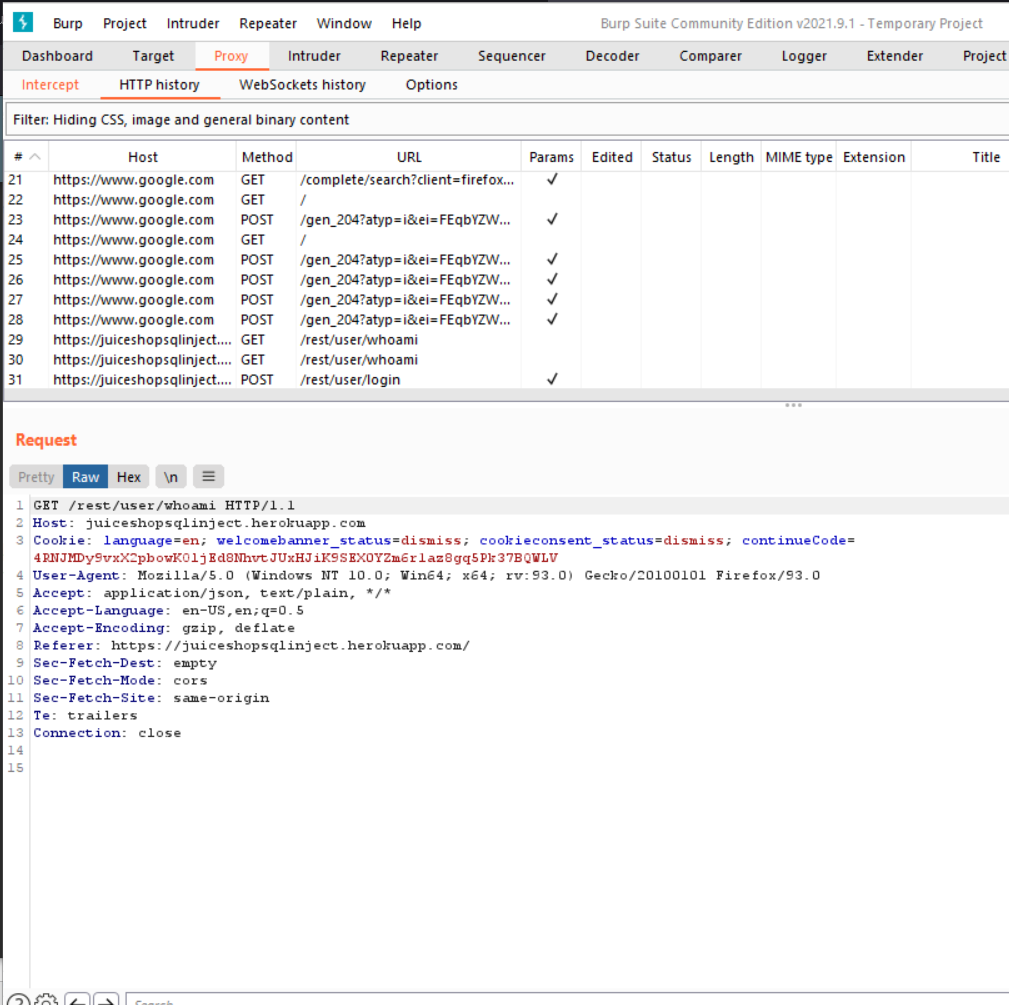
**XSS Attack on the Web Application**

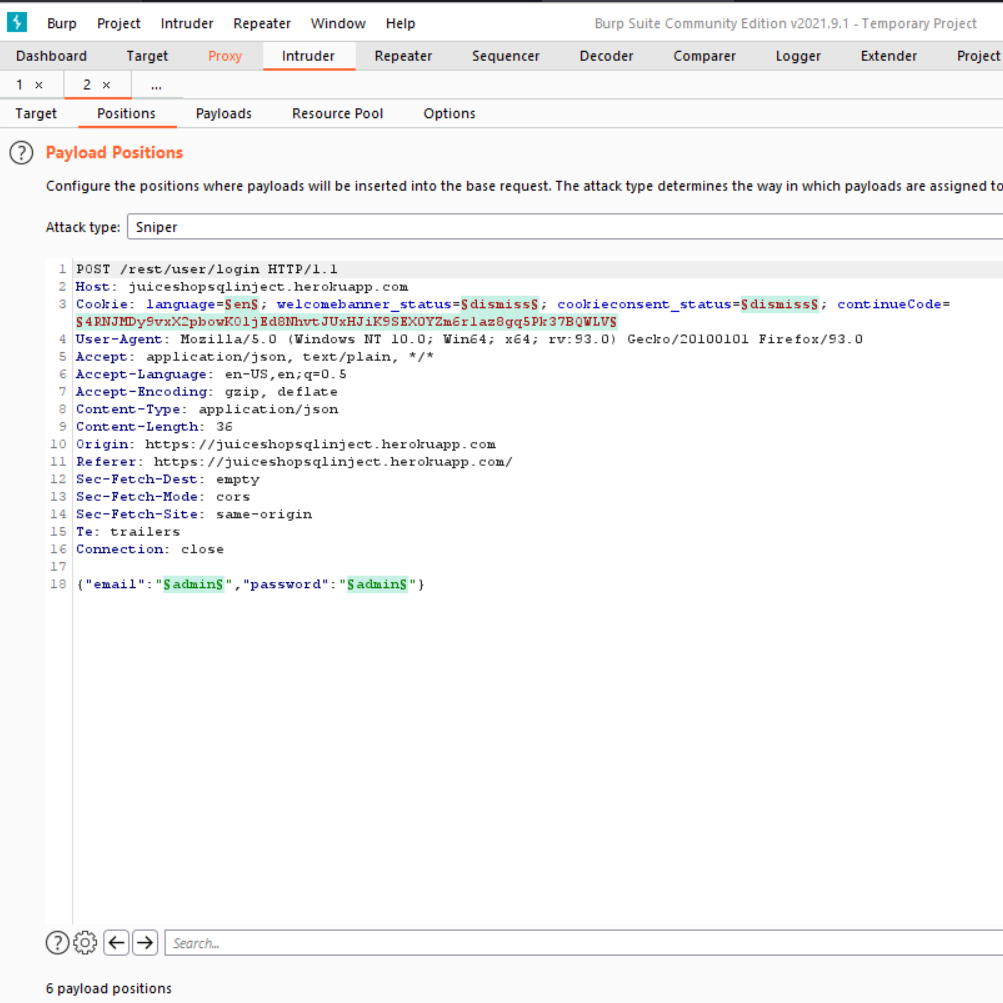
**Performing DOM XSS attack by using the code <iframe src="javascript:alert(`DOM xss Attack on your application`)">.Similarly various commands can be executed in the application when there is no protection against XSS payload.**



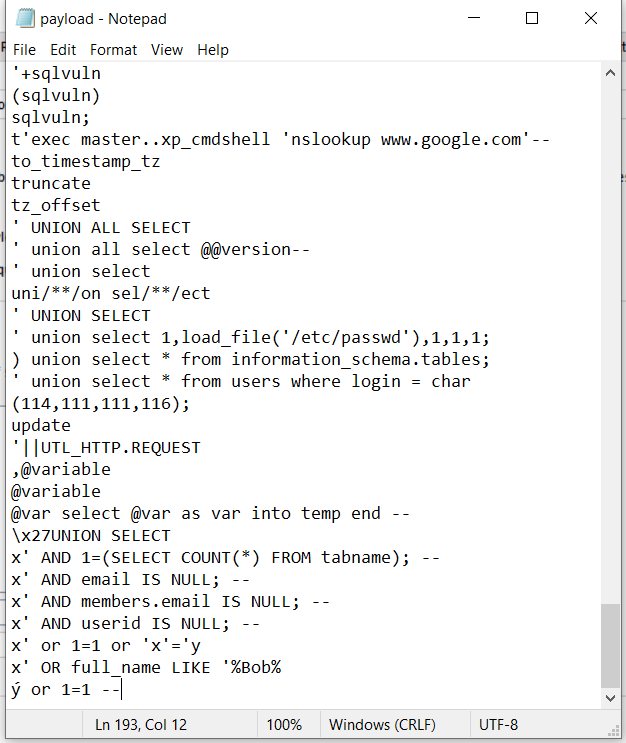
Solution: The best way to fix DOM based cross-site scripting is to use the right output method (sink). For example if you want to use user input to write in a div tag element don't use innerHtml, instead use innerText or textContent. This will solve the problem, and it is the right way to re-mediate DOM based XSS vulnerabilities.

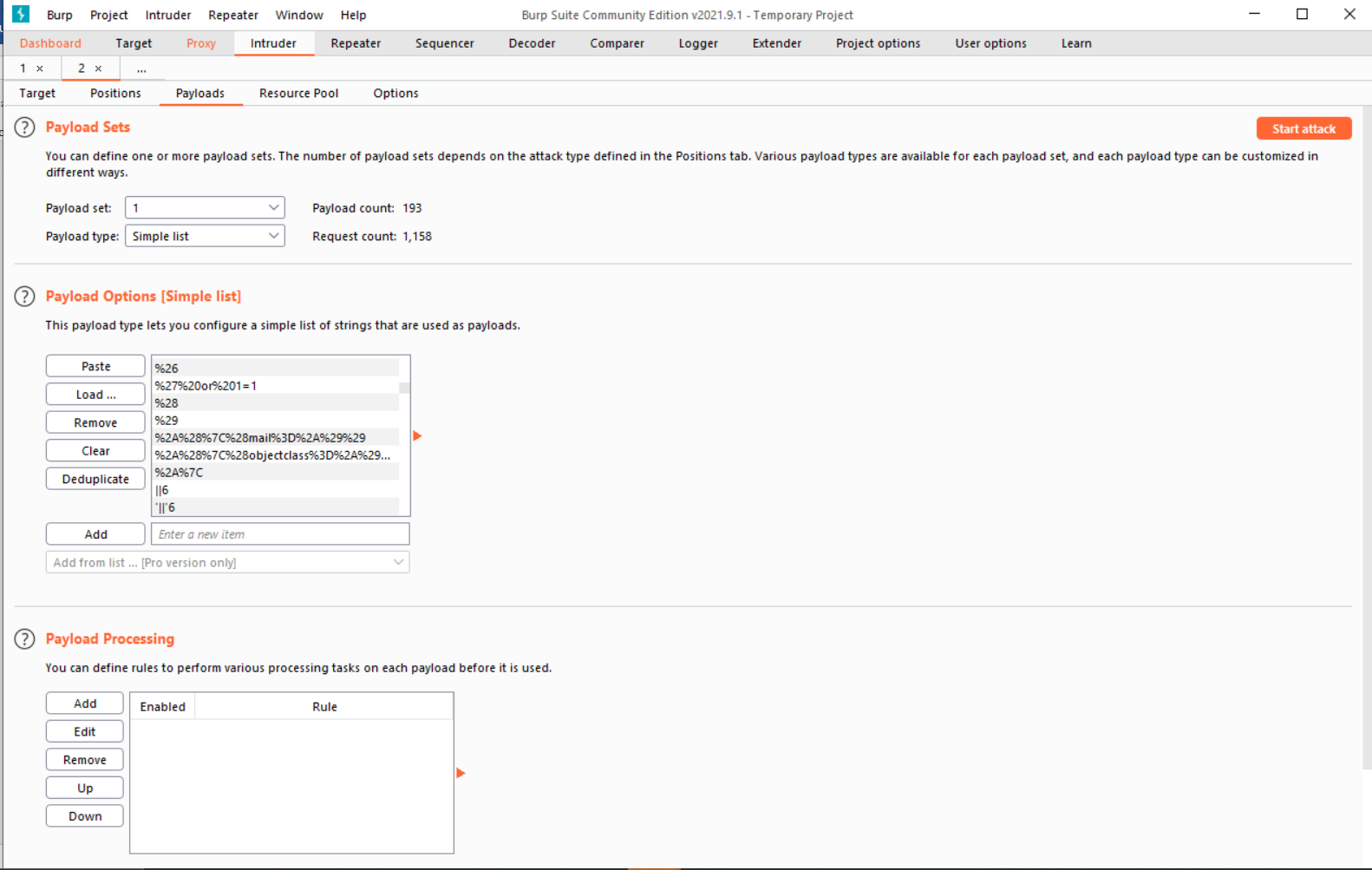
SQL Injection attack on the Login Page using Burp Suite



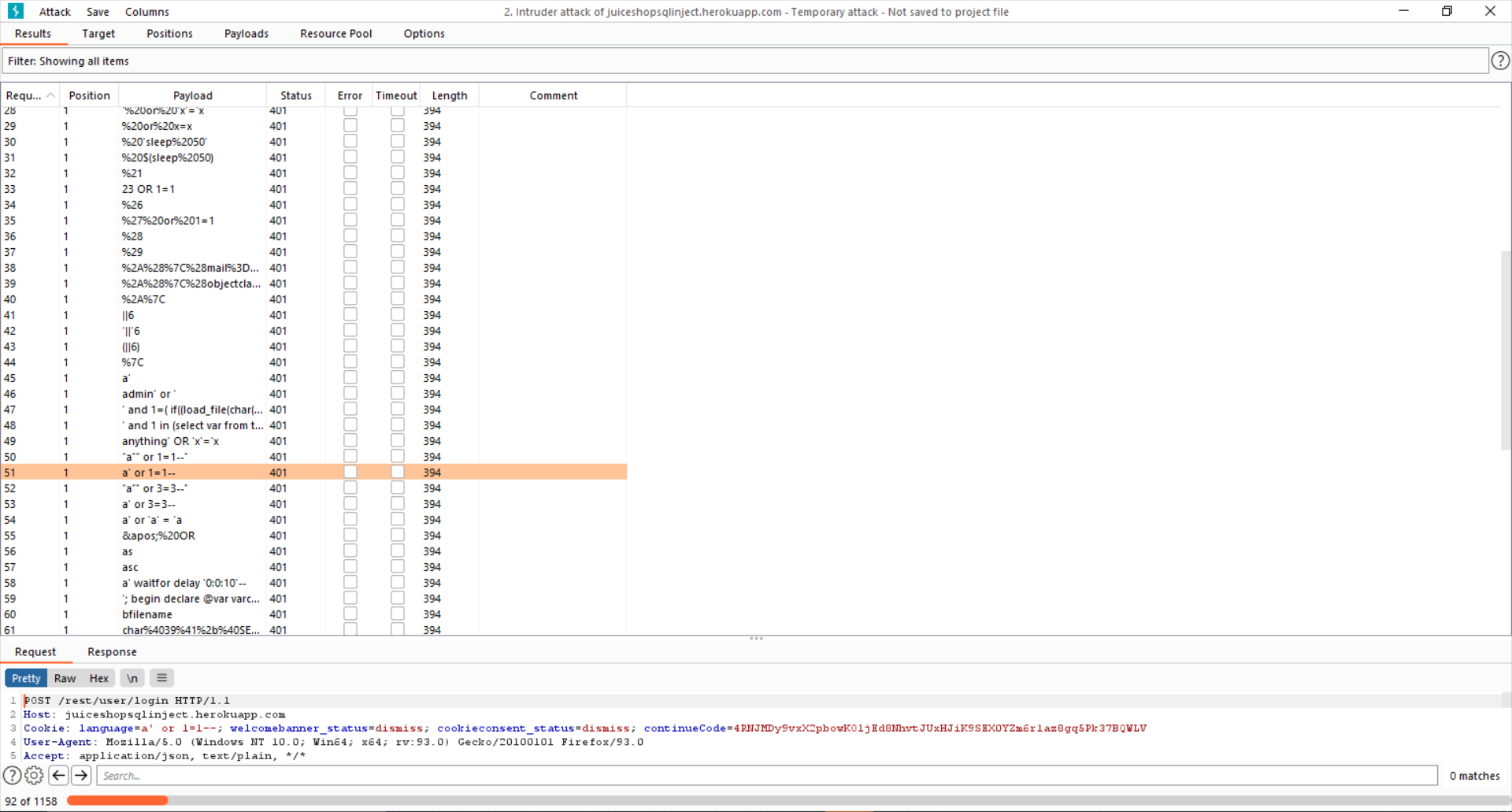


**Creating a new payload.txt for attacking the login page to gain access as Admin Account**

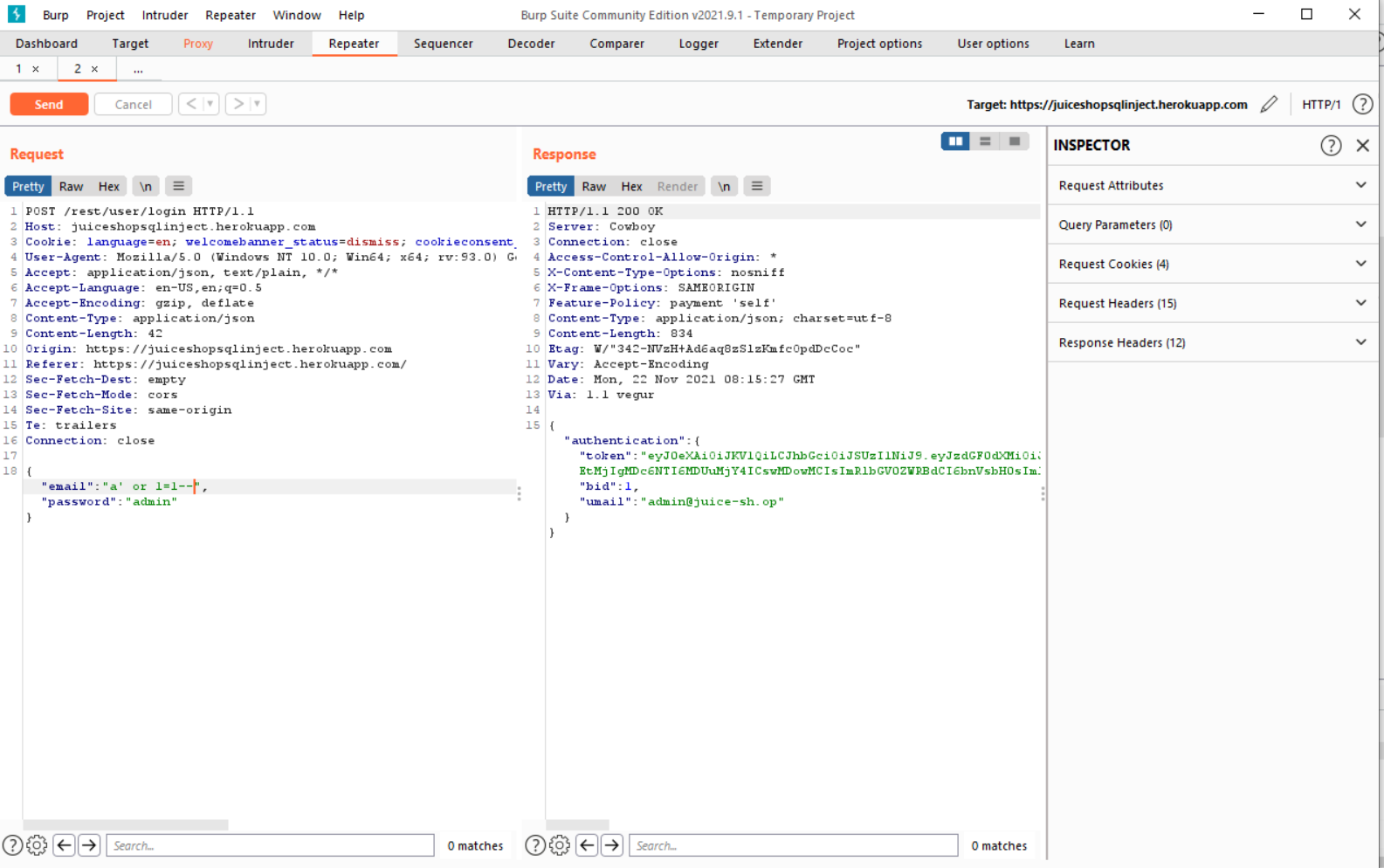


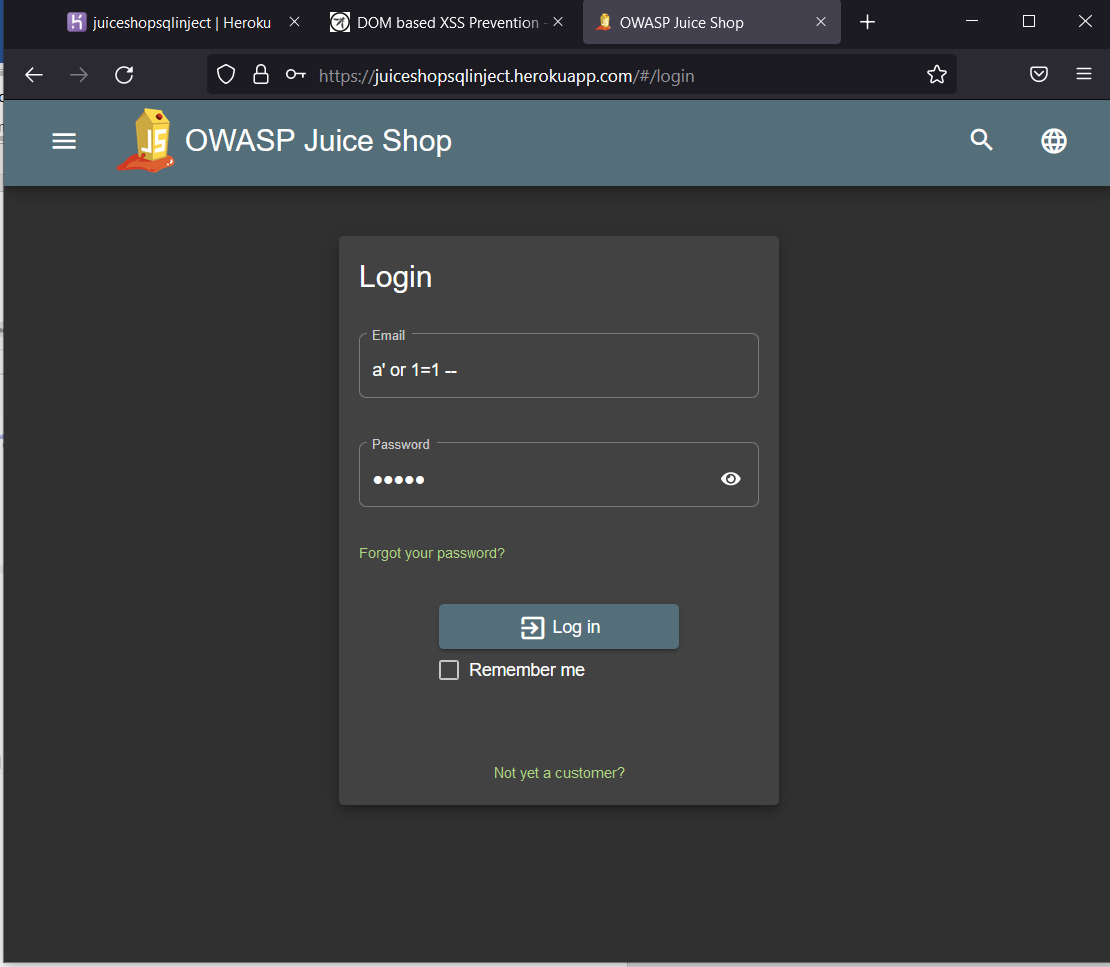


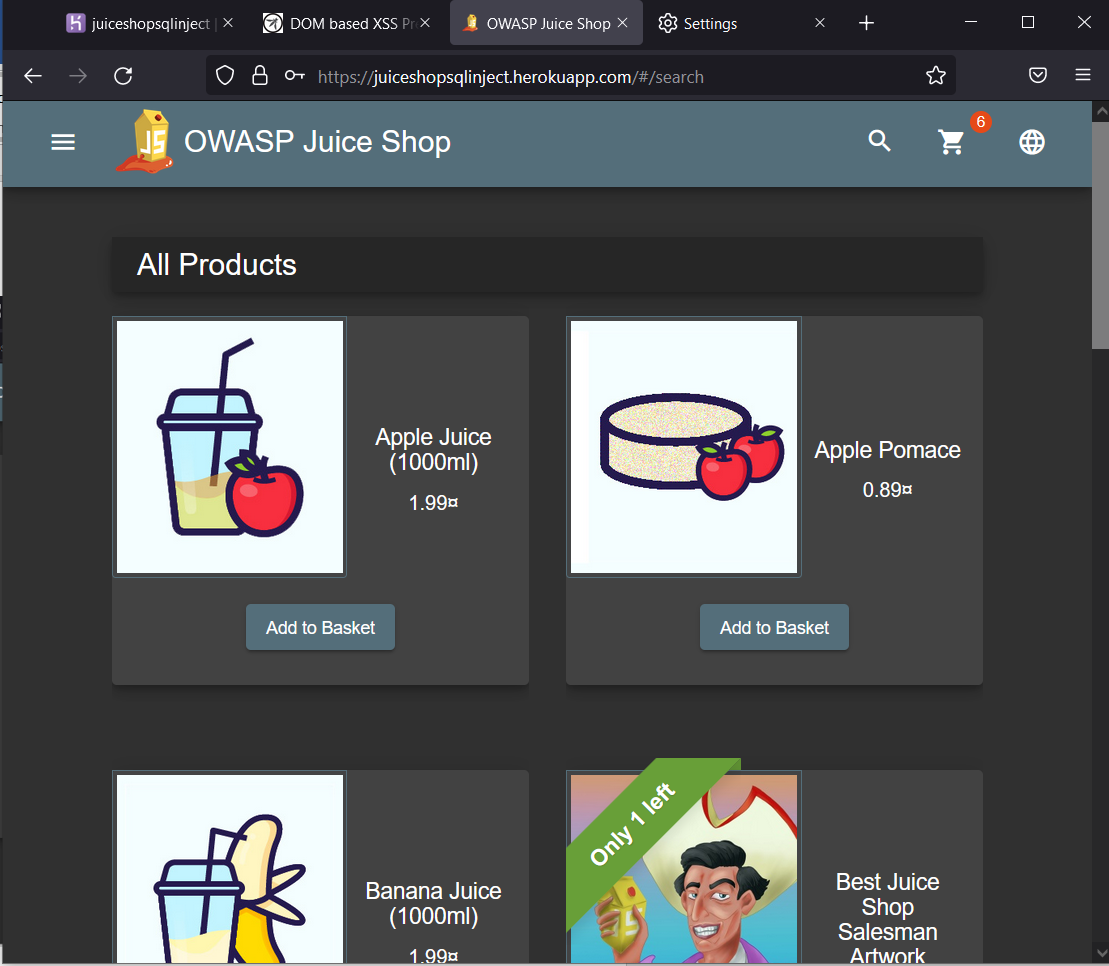
**Then start the attack using the Intruder attack we found the correct SQL injection command to be inserted .**



**Inserting the command through by forwarding the request to the Repeater**



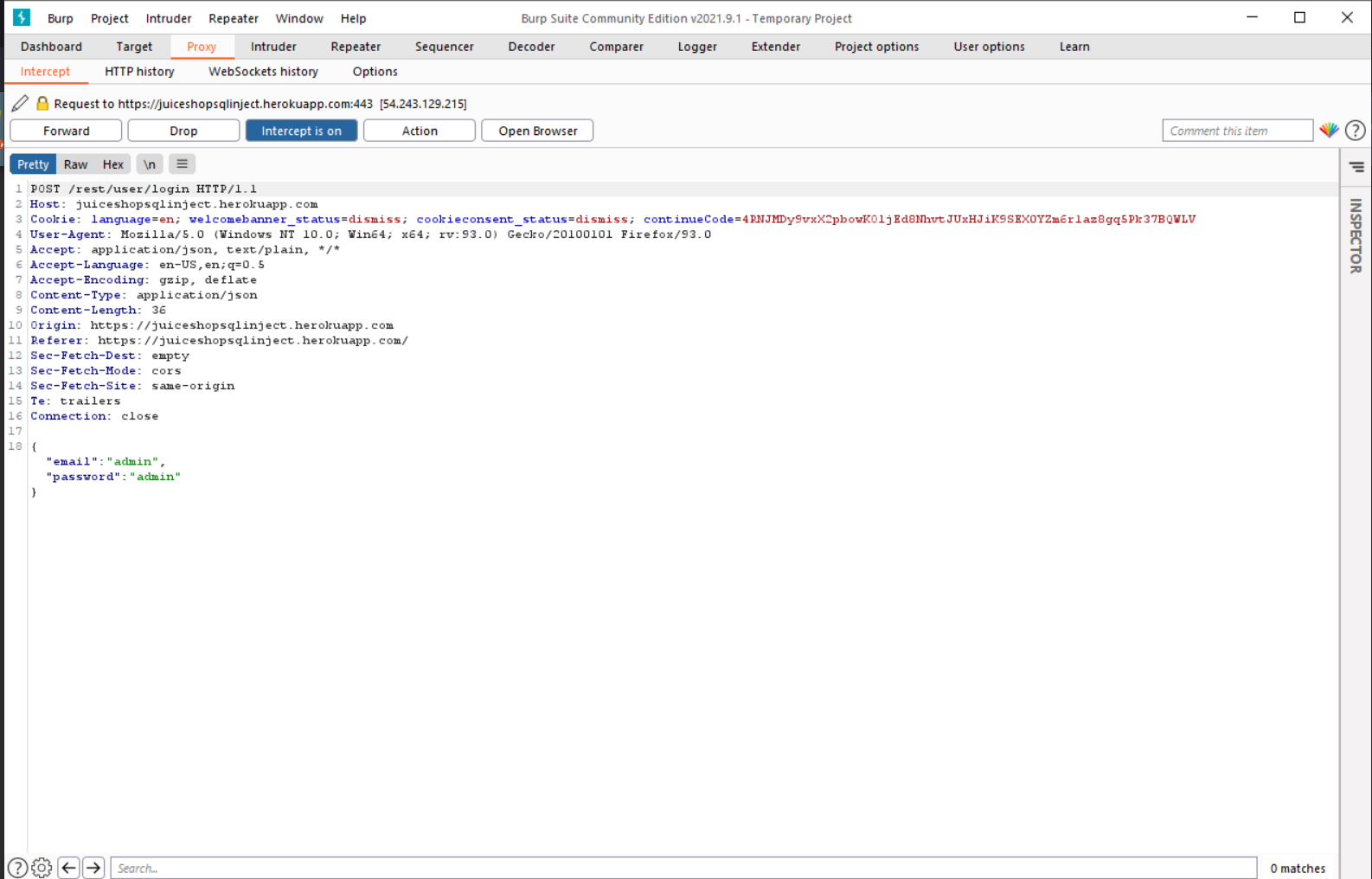


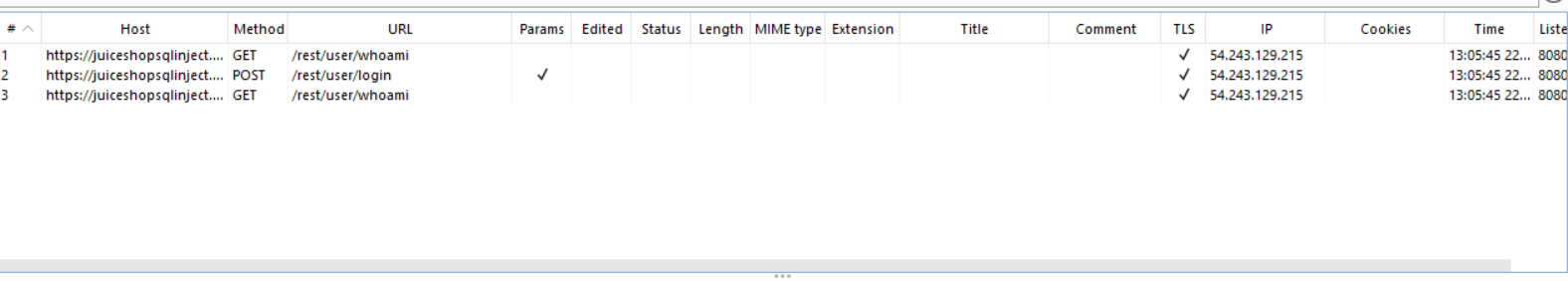
**Finally login as Admin Account**

**Another method of randomly inserting SQL injections**

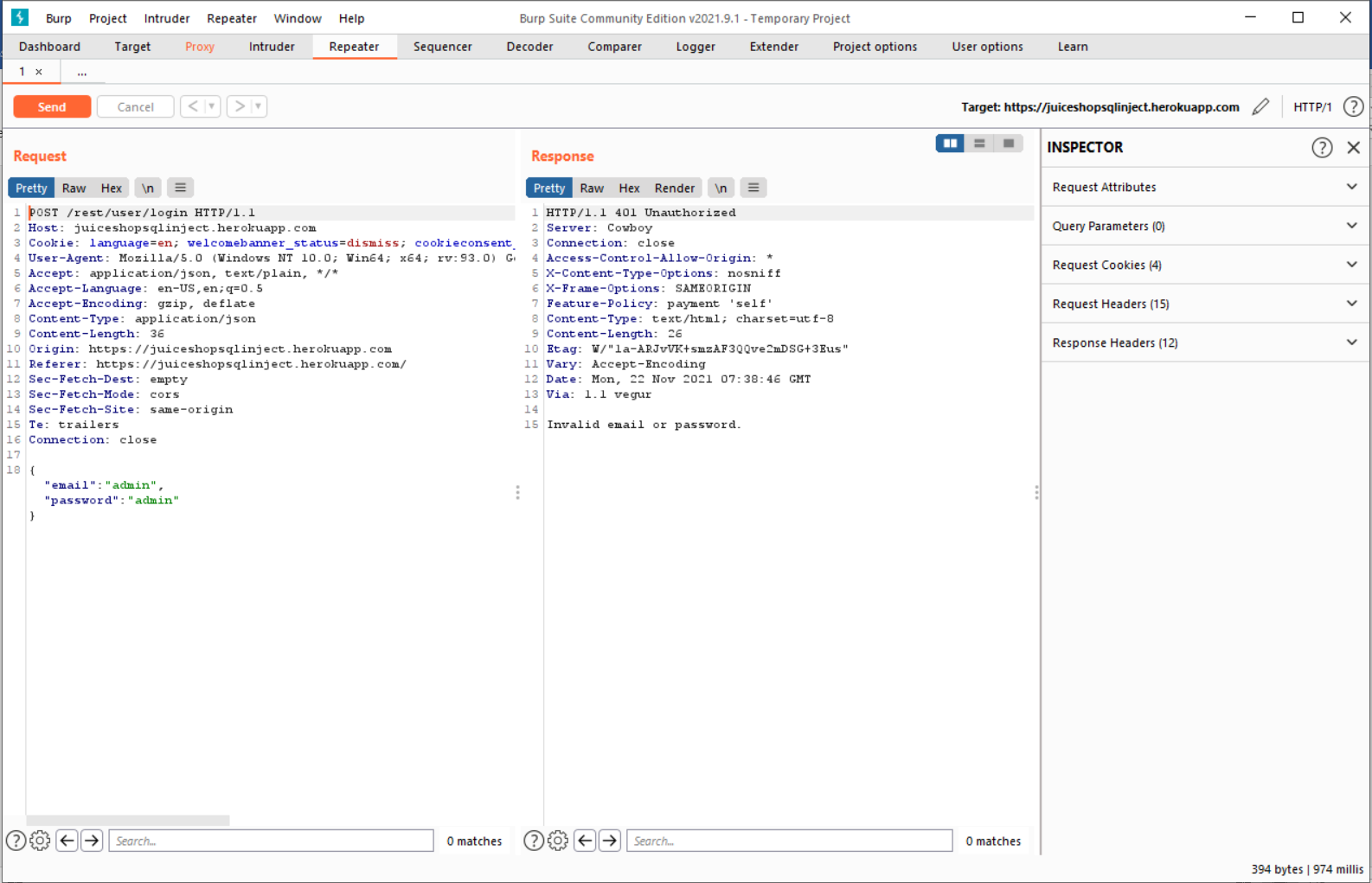
Login as Admin

After entering some random email and password





Sending the request to repeater



After SQL injection command

We get the authentication successfully

