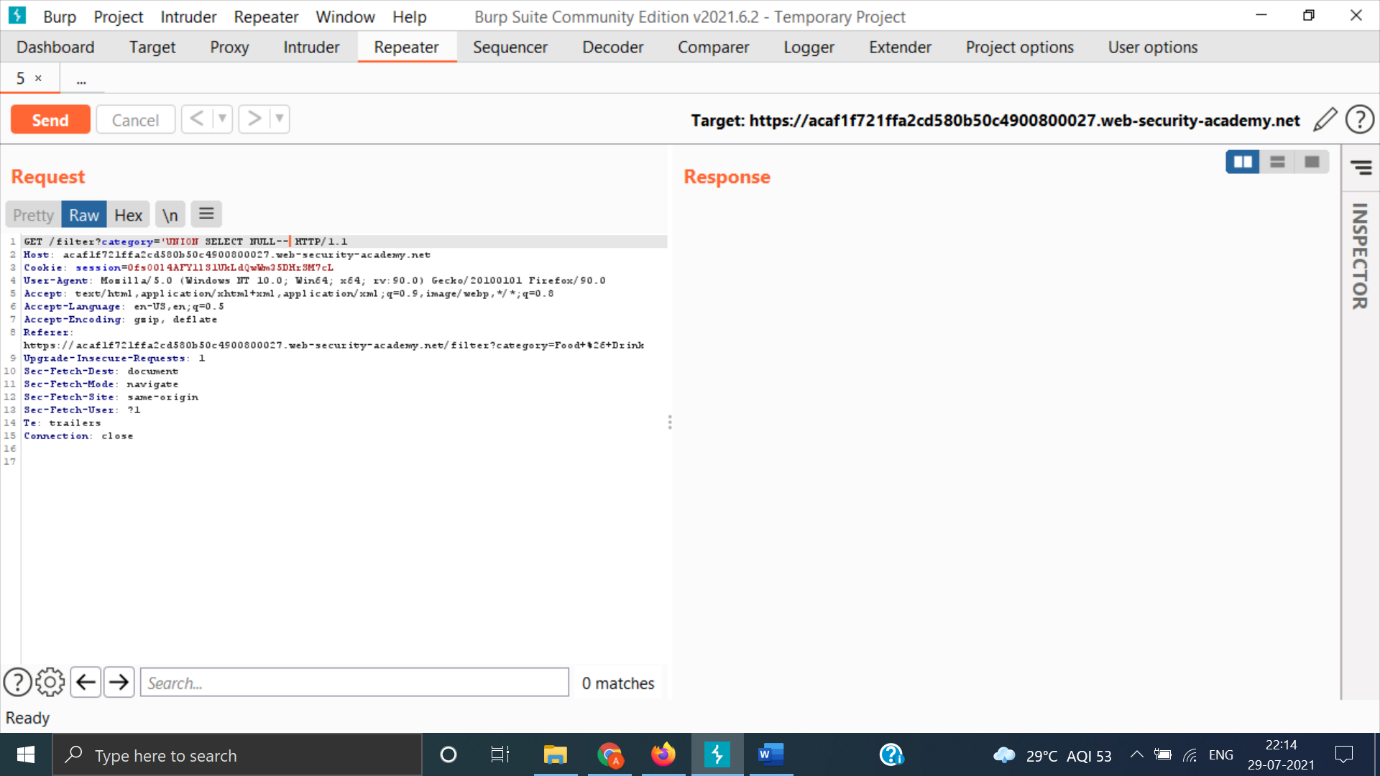
**UNION Based SQL injection**

**Lab: SQL injection UNION attack, determining the number of columns returned by the query**

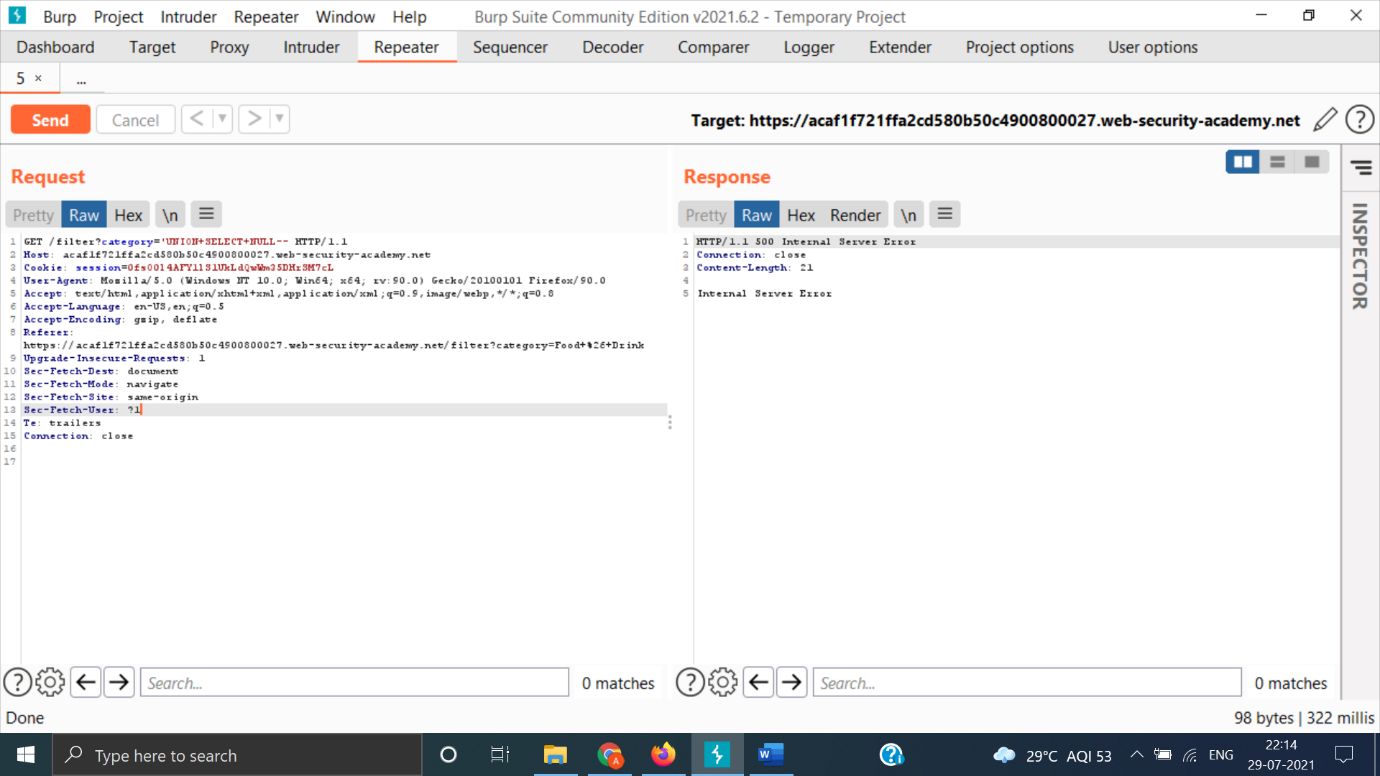
This lab contains an SQL injection vulnerability in the product category filter. The results from the query are returned in the application's response, so you can use a UNION attack to retrieve data from other tables. The first step of such an attack is to determine the number of columns that are being returned by the query. You will then use this technique in subsequent labs to construct the full attack. To solve the lab, determine the number of columns returned by the query by performing an [SQL injection UNION](https://portswigger.net/web-security/sql-injection/union-attacks) attack that returns an additional row containing null values.

Method 1:

Basic payload used: UNION SELECT NULL--

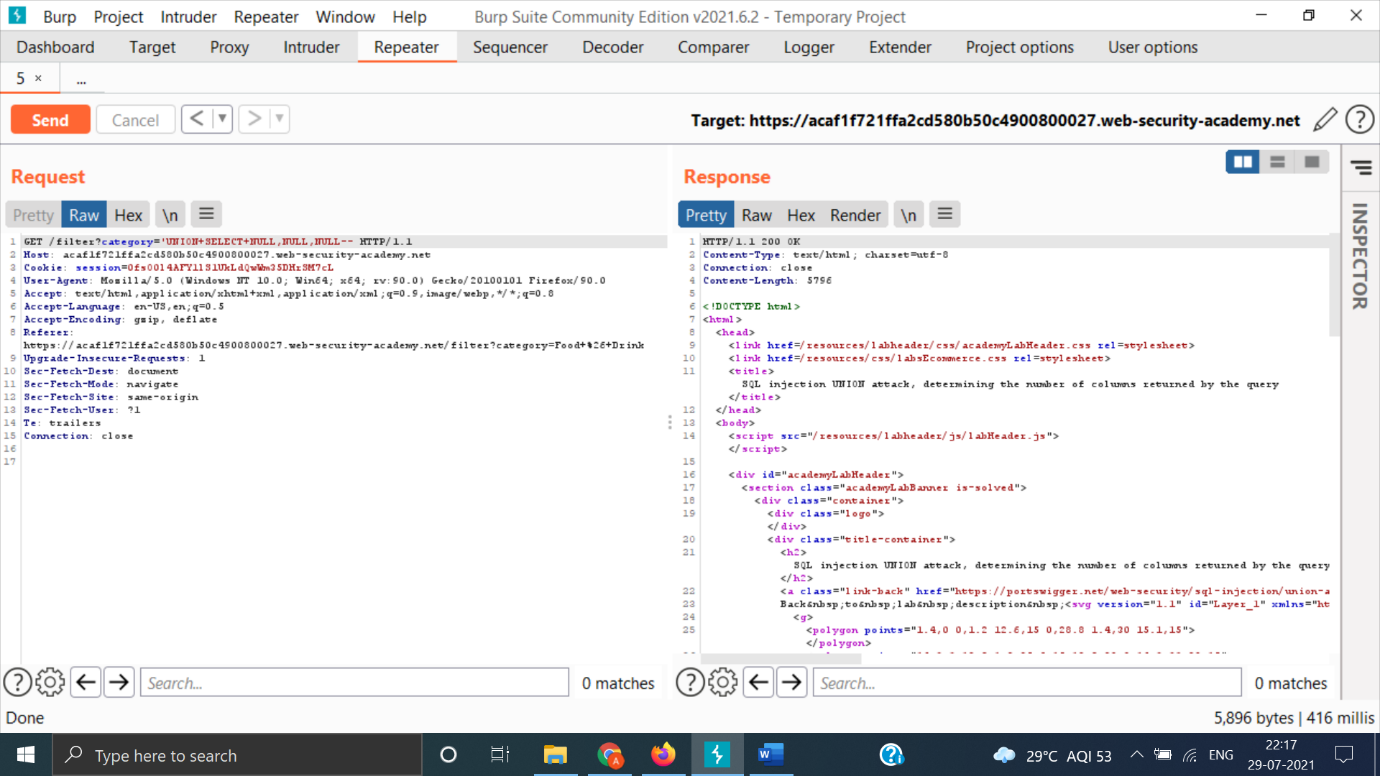


The payload is not URL encoded. Encode it by selecting the payload and pressing Ctrl+U.



By this we found that possibility of 1 column for both queries is not true.

Lets increment the query.

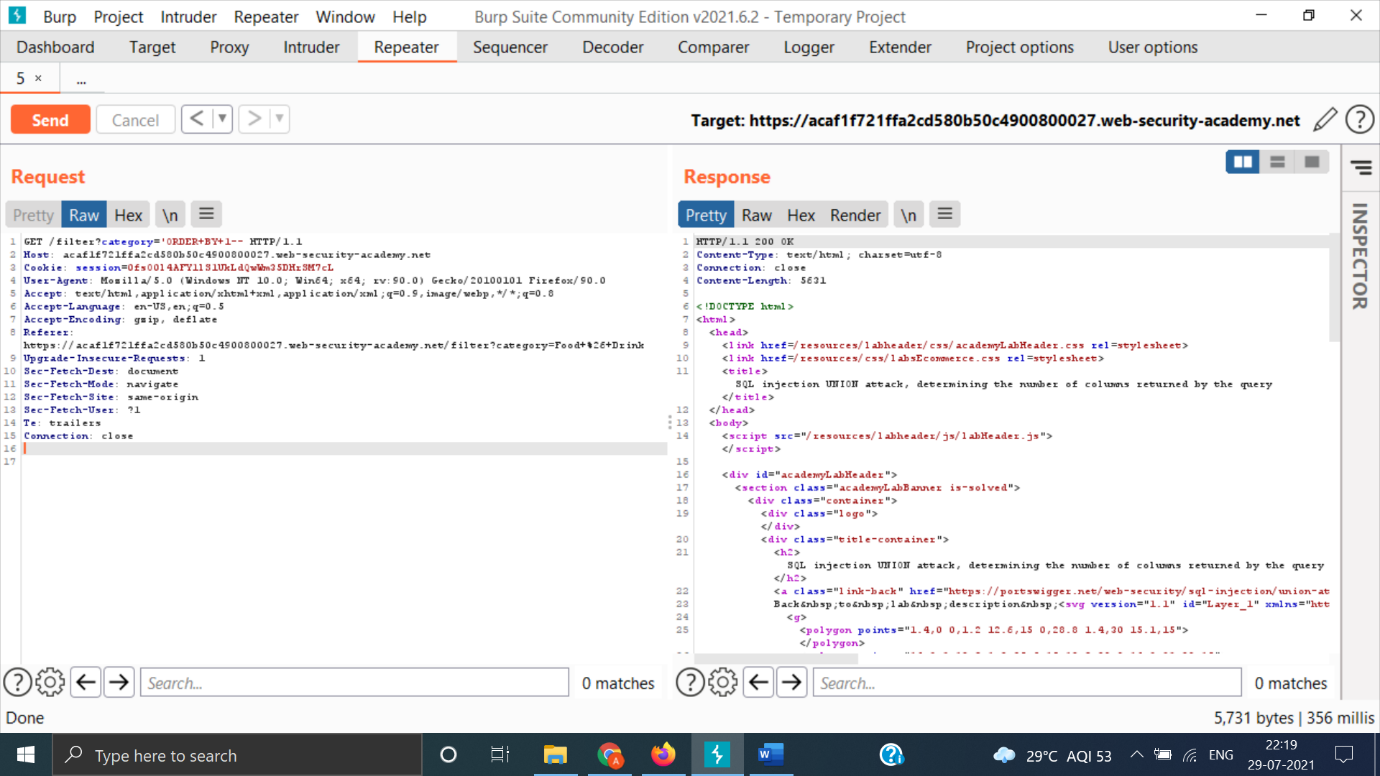


On using 3 NULL parameters we got HTTP 200 status code telling us that there are a total of 3 columns.

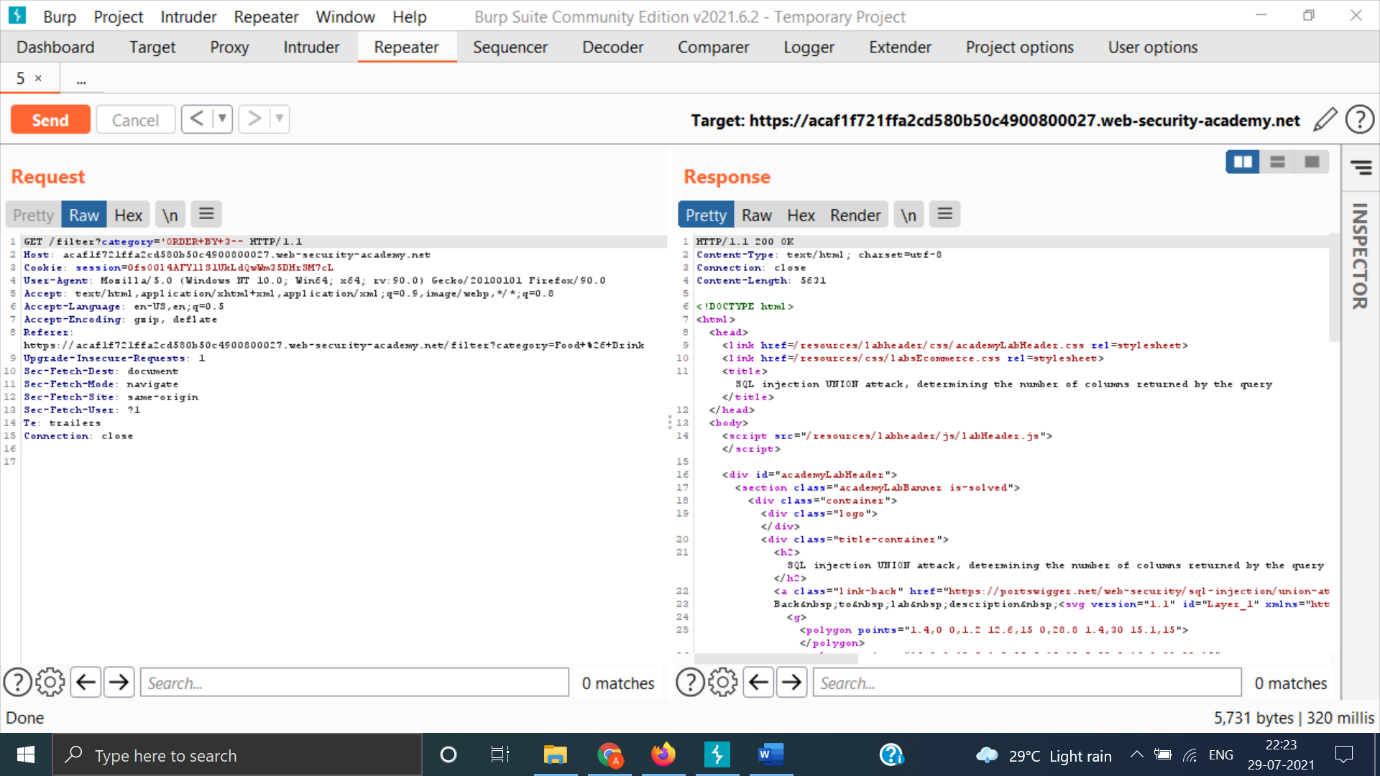
Method 2:

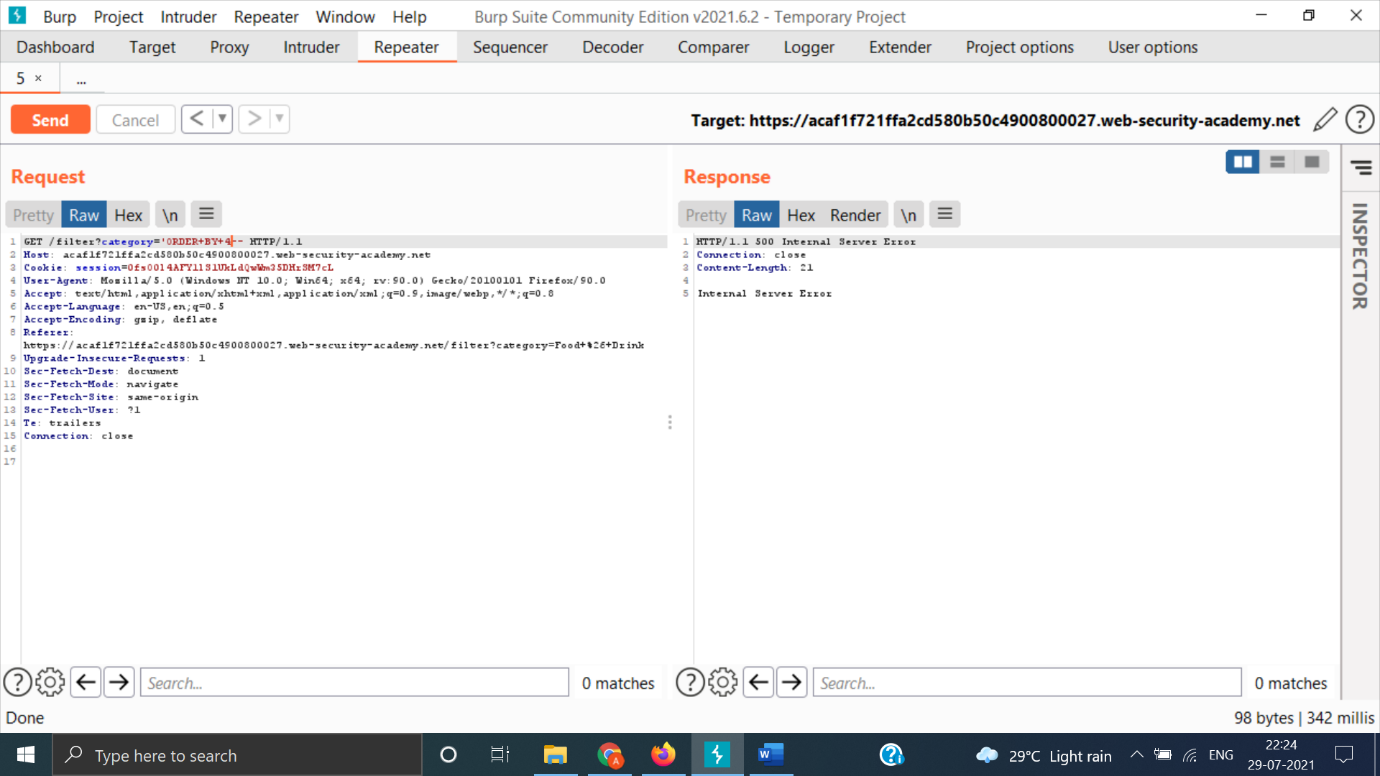
Basic payload used: ‘ORDER BY 1--

So, this method can be used to determine number of columns, as we increase the value from 1 to other integer in above payload at some stage, we will not be getting HTTP 200 status code. The minimum integral value will be “m” and the number of columns will be “m-1”.



So, now we will keep on incrementing the integral value.





So, we found that m=4 and number of columns are m-1 i.e., 3.

So, by using both methods we found that number of columns are 3.