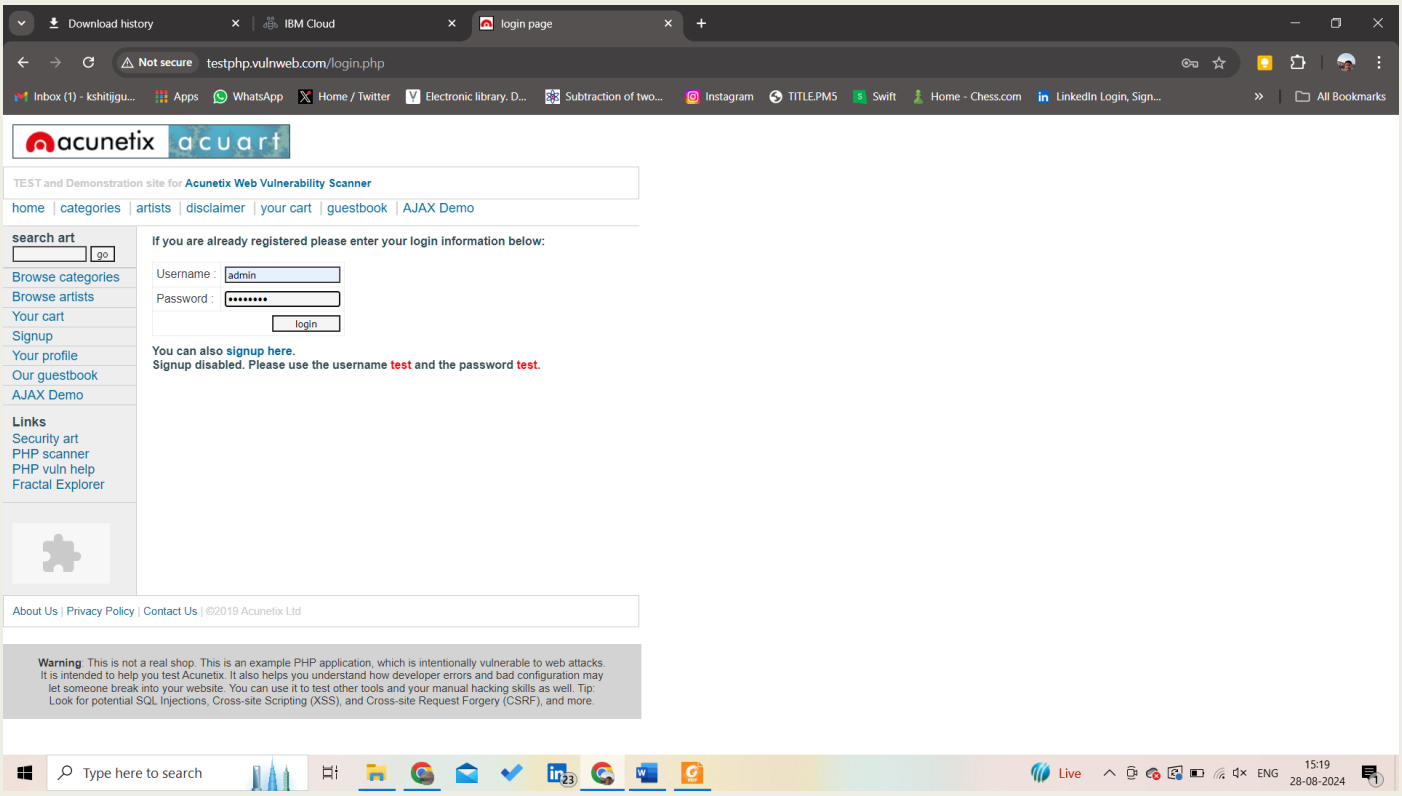
NAME: KSHITIJ GUPTA
Enrolment Number: 21162101007
Sub: CS

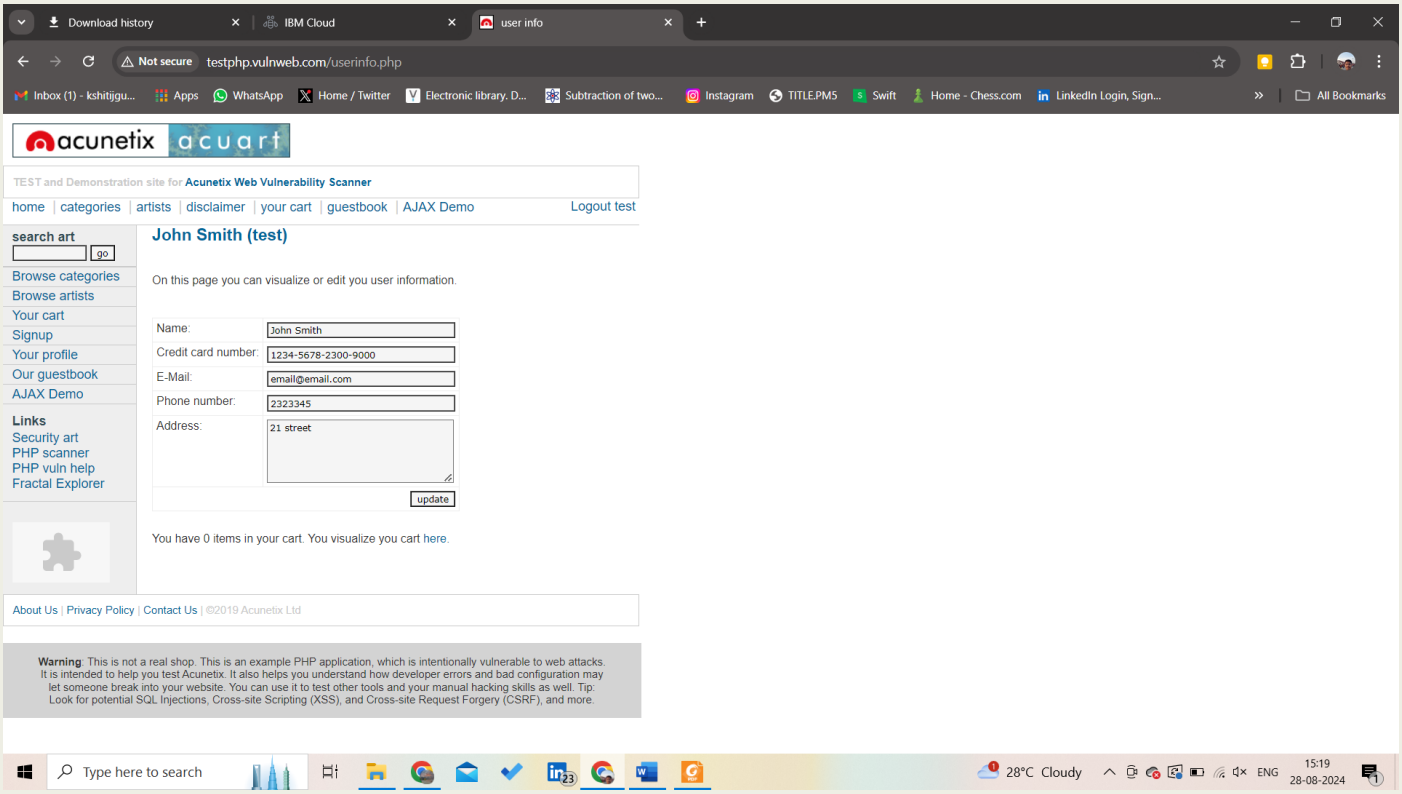
Practical – 2[Batch-71]

You are a cloud security analyst for ane-commerce website (testphp.vulnweb.com), and your task is to perform a security assessment of their online store. During the assessment, you discover a potential vulnerability in their functionality, which is susceptible to a Union-based SQL injection attack.

Exploit the functionality of the e-commerce website to bypass the login page as well as retrieve sensitive information from the database.

First vulnerability we exploit is in the Login. We put the username as admin and password as password‘or‘1’=’1

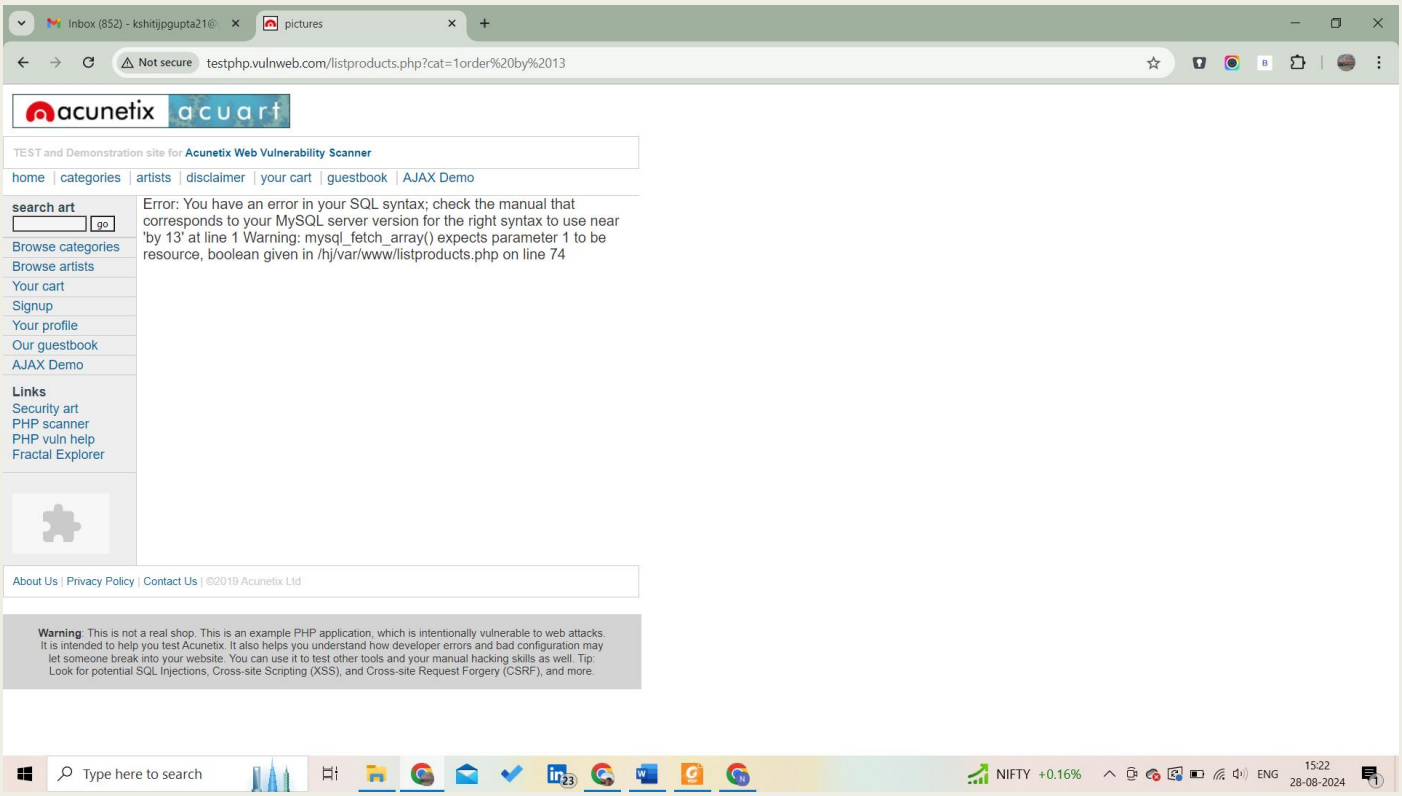




This attack is an SQL injection where the input password' or '1' = '1 always evaluates to true, bypassing authentication. To prevent it, use parameterized queries or prepared statements. Additionally, validate and sanitize all user inputs.

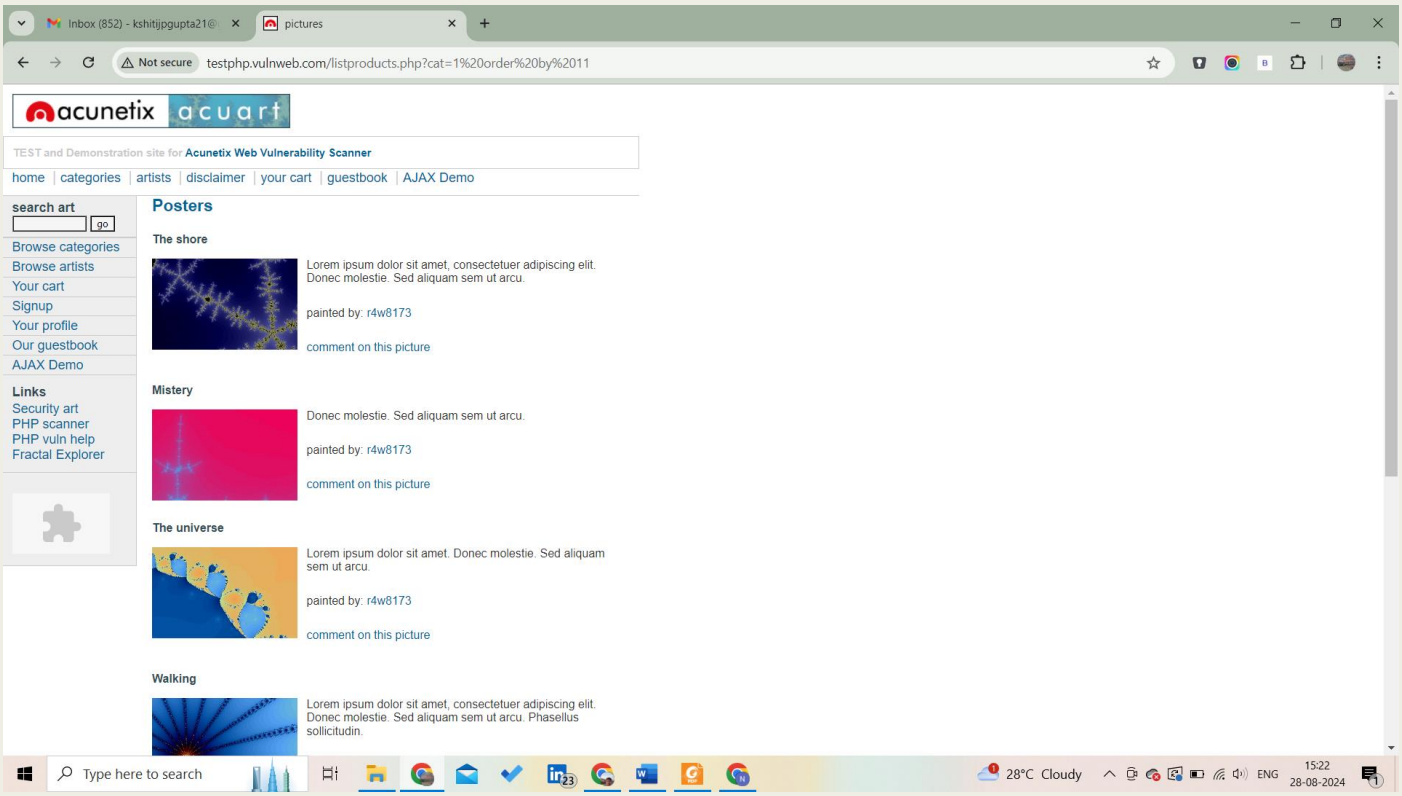
Now we are retrieving the number of columns using order by. For this go to the Categories tab and click on any category, now we add the following URL:

http://testphp.vulnweb.com/listproducts.php?cat=1order%20by%2013



As we can see there is no product, now this is a hit and trial method. We will now change URL to :

http://testphp.vulnweb.com/listproducts.php?cat=1%20order%20by%2011



To determine the number of columns in a table using the `ORDER BY` clause, you can increment the column index in the URL until you receive an error. For example, start with `order by 1`, then `order by 2`, and so on. When you reach a number that causes an error, the previous number is the total count of columns.

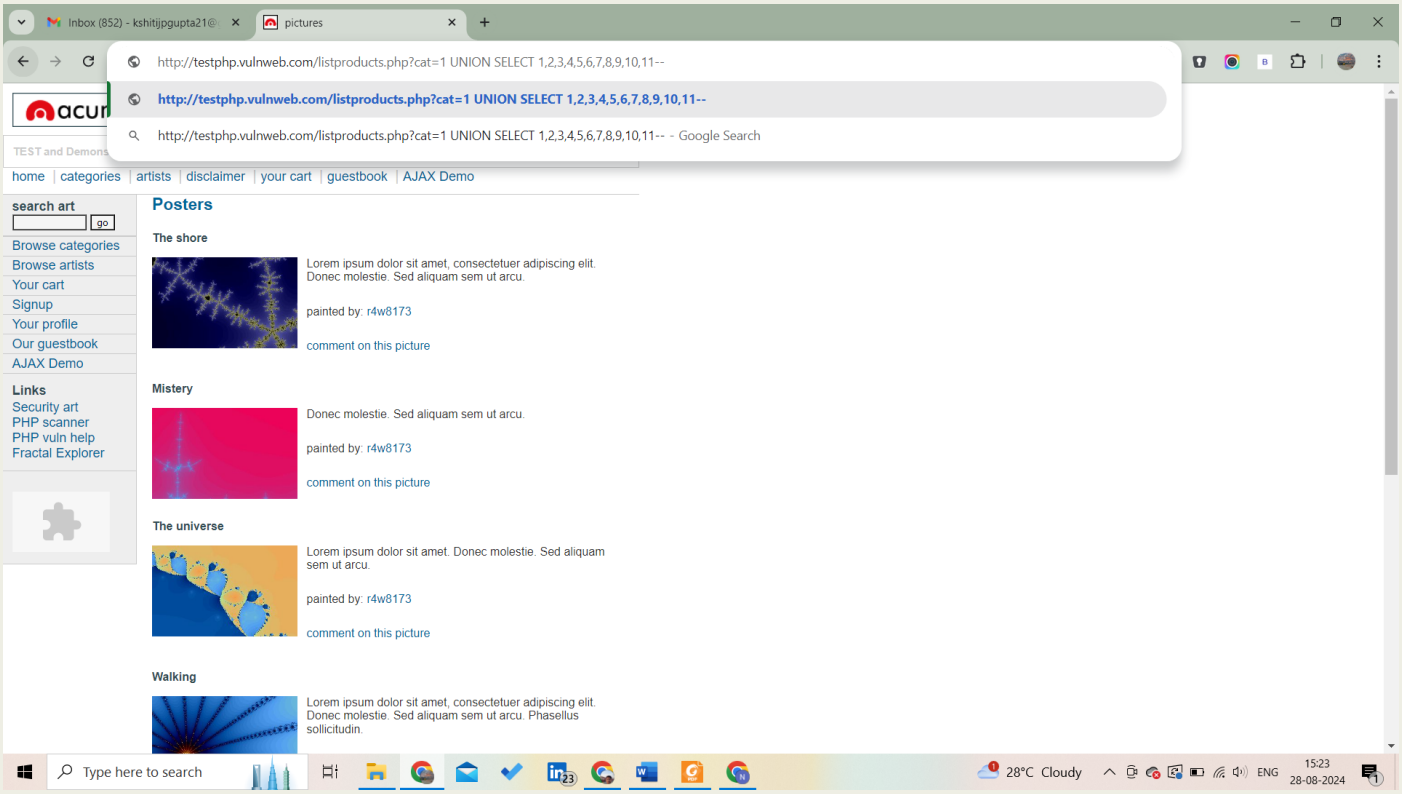
To solve this problem use parameterized queries or prepared statements to prevent SQL injection attacks.

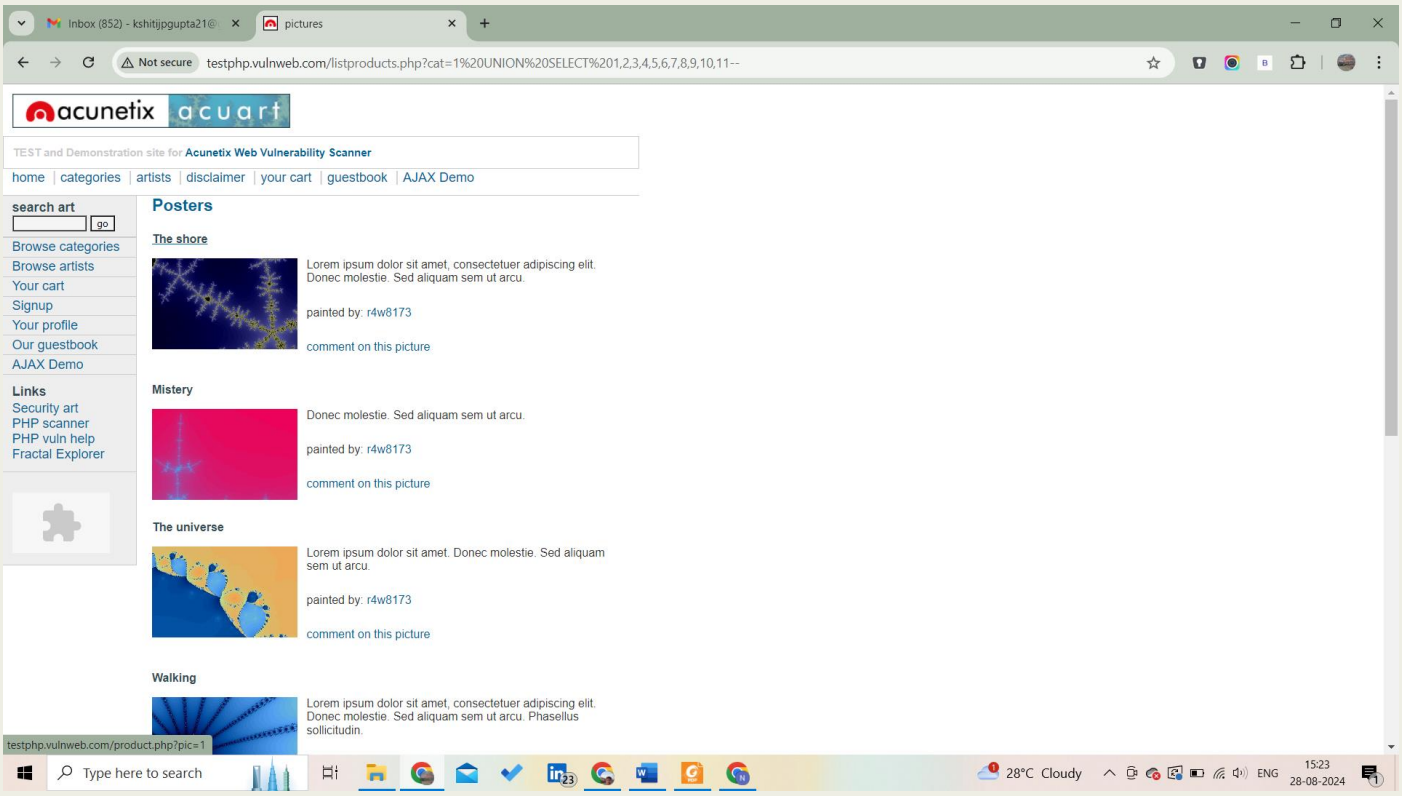
Now we are going to retrieve the injectable columns to do this we will use

this URL:

http://testphp.vulnweb.com/listproducts.php?cat=1 UNION SELECT

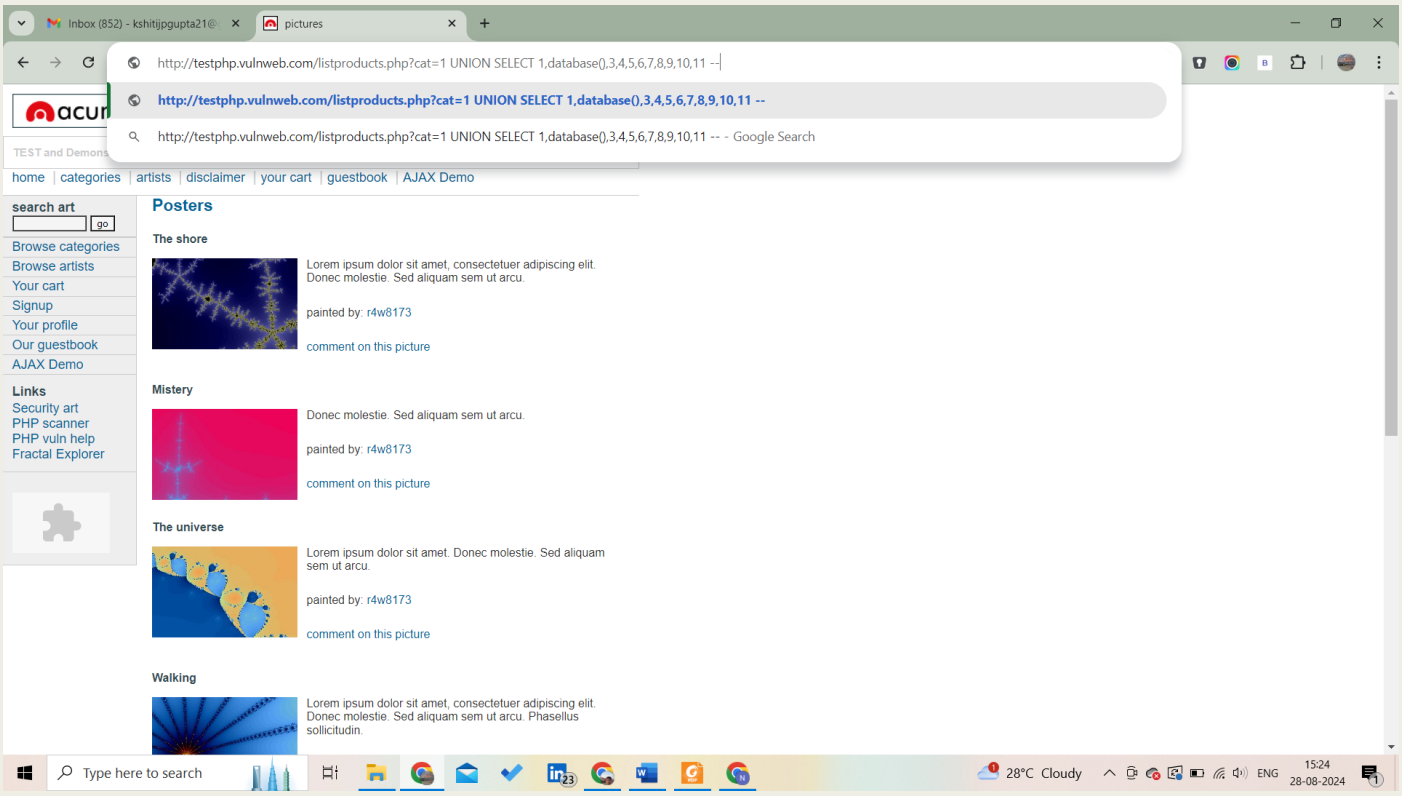
1,2,3,4,5,6,7,8,9,10,11—

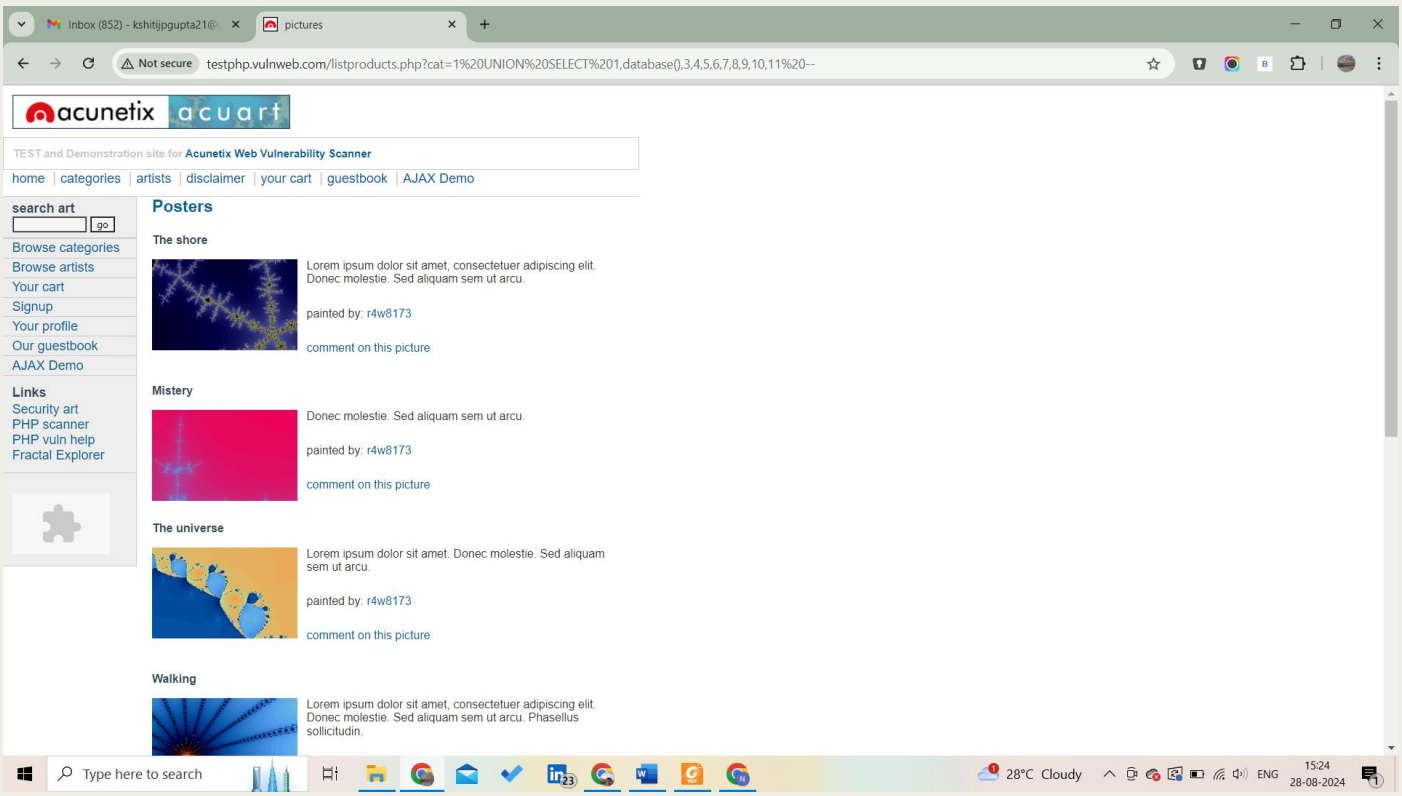




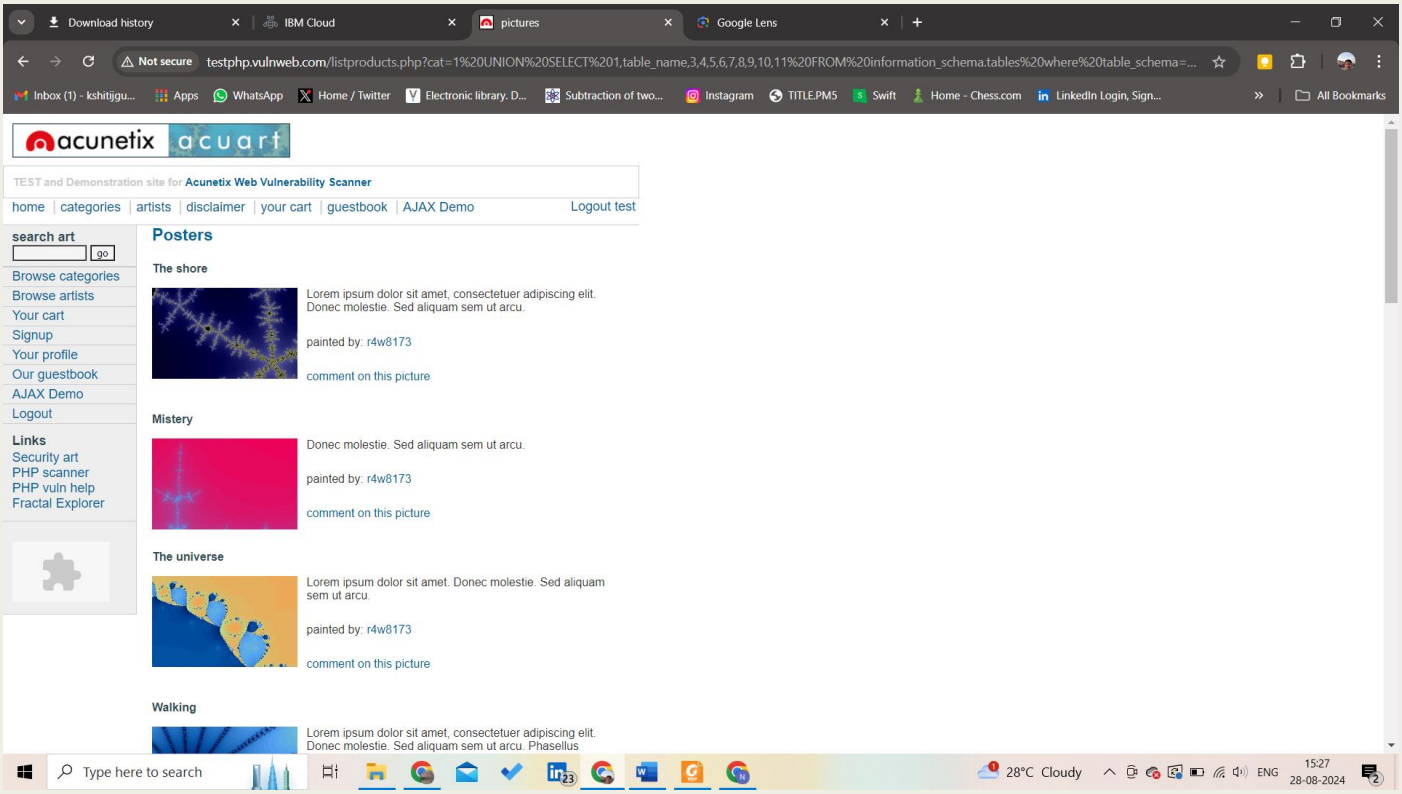
By using the UNION SELECT statement, you can combine the results of two or more SELECT queries. You incrementally add columns (e.g., 1,2,3,...) until the query executes without an error. This helps identify the exact number of columns in the table.

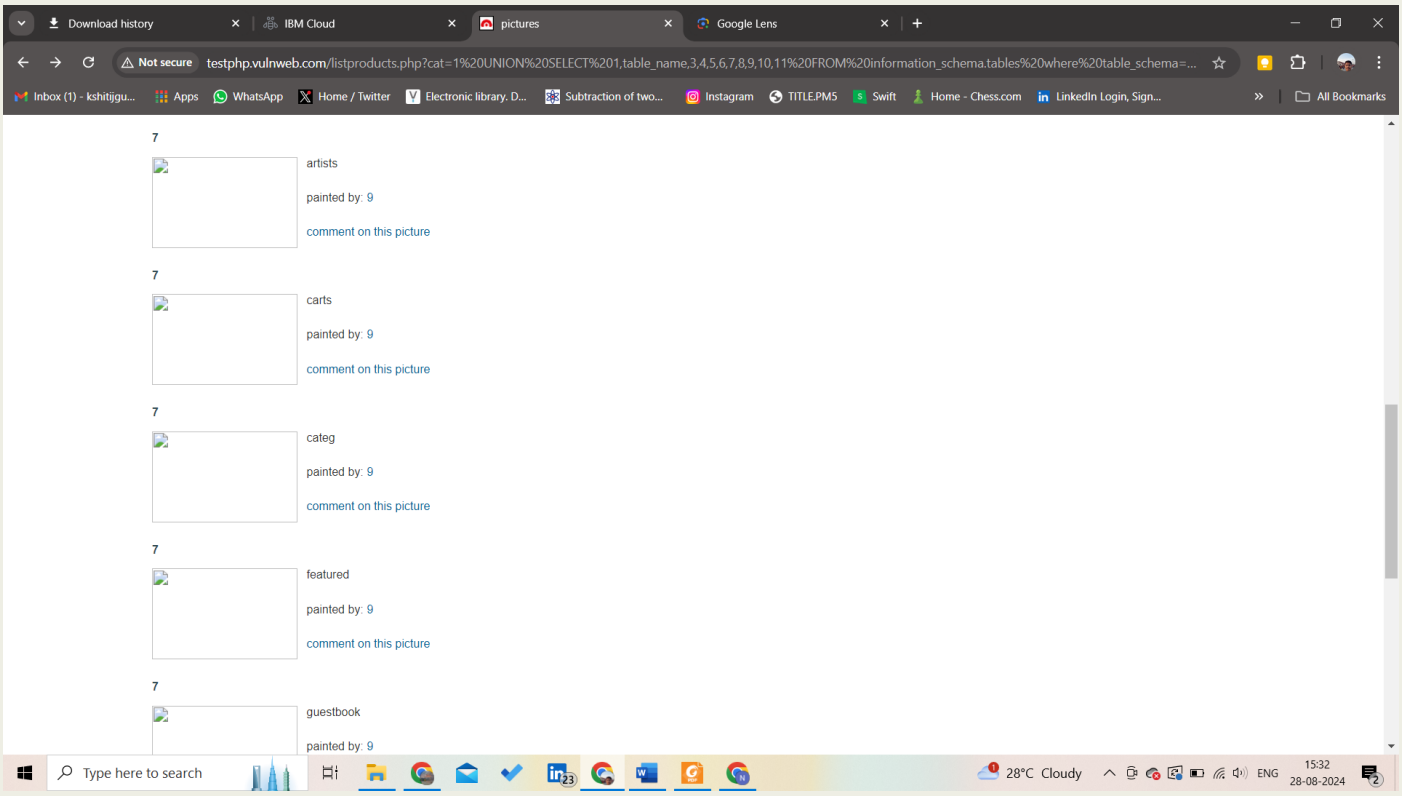
Now we want to find the database name by replacing any one of the columns with database() hence the URL will be: http://testphp.vulnweb.com/listproducts.php?cat=1 UNION SELECT 1,database(),3,4,5,6,7,8,9,10,11—



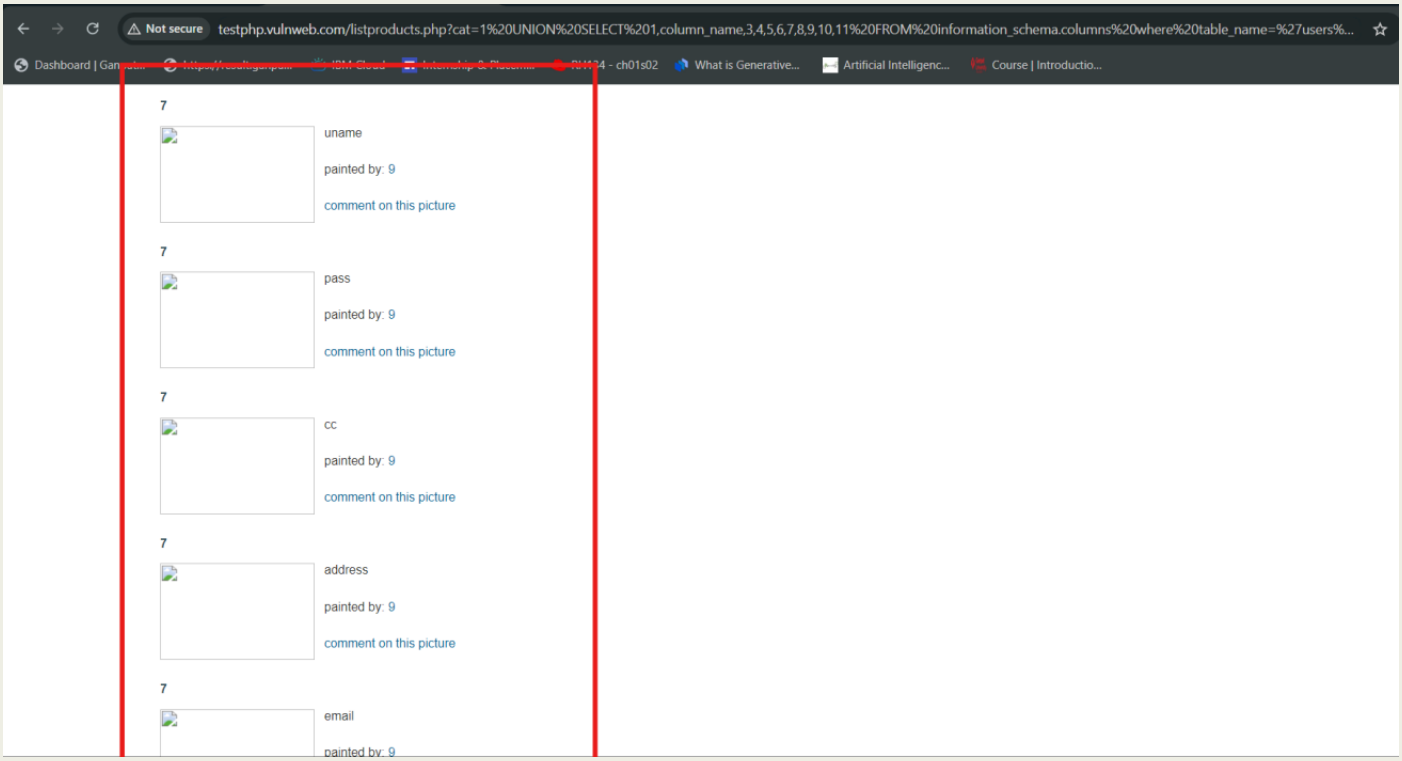


Now we will be trying to retrieve the tables names here we will add the table_name in one of the columns in URL along with the FROM information_schema.tables where table_schema='acuart'—

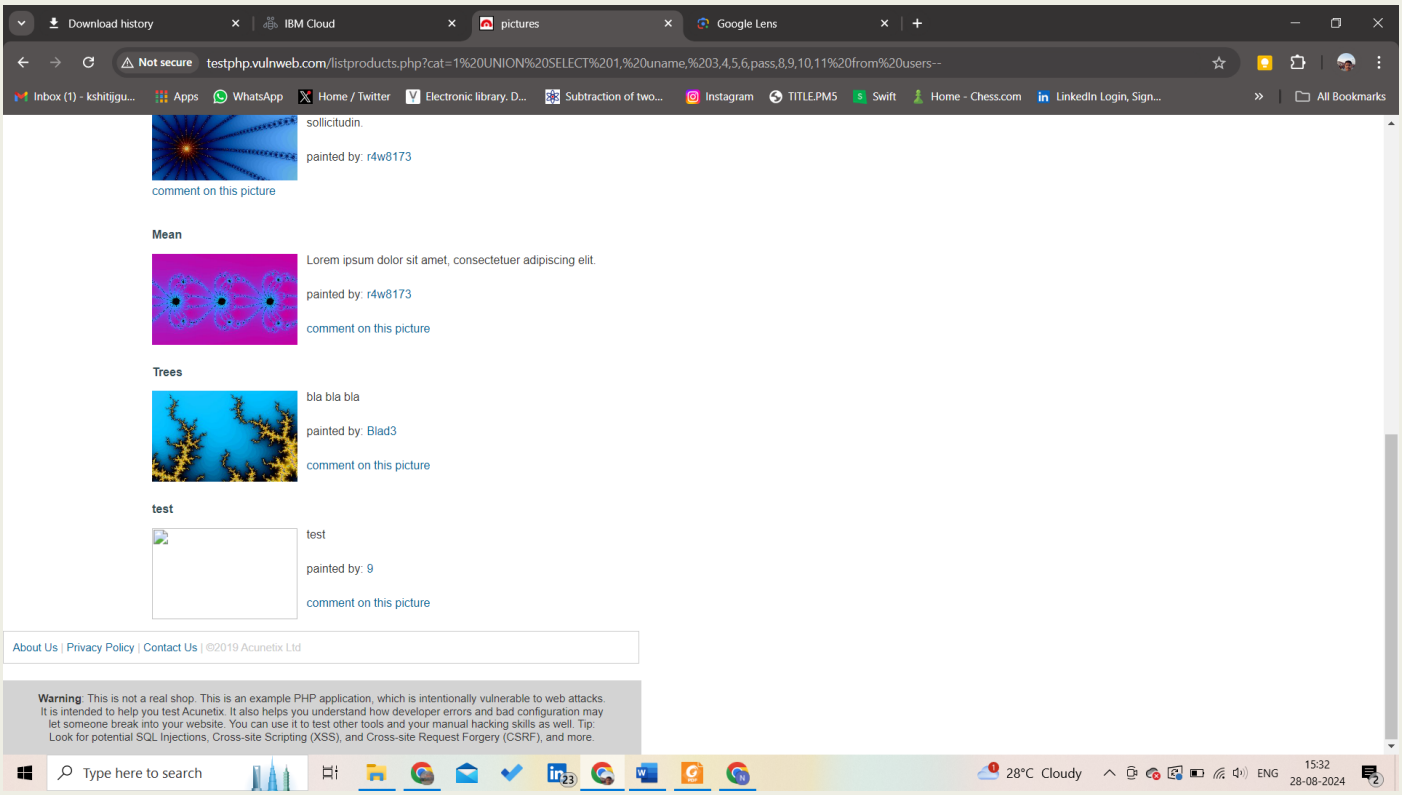




Now we will try to find out the columns of any table where we will add column_name in one of the columns along with FROM information_schema.columns where table_name='users'—

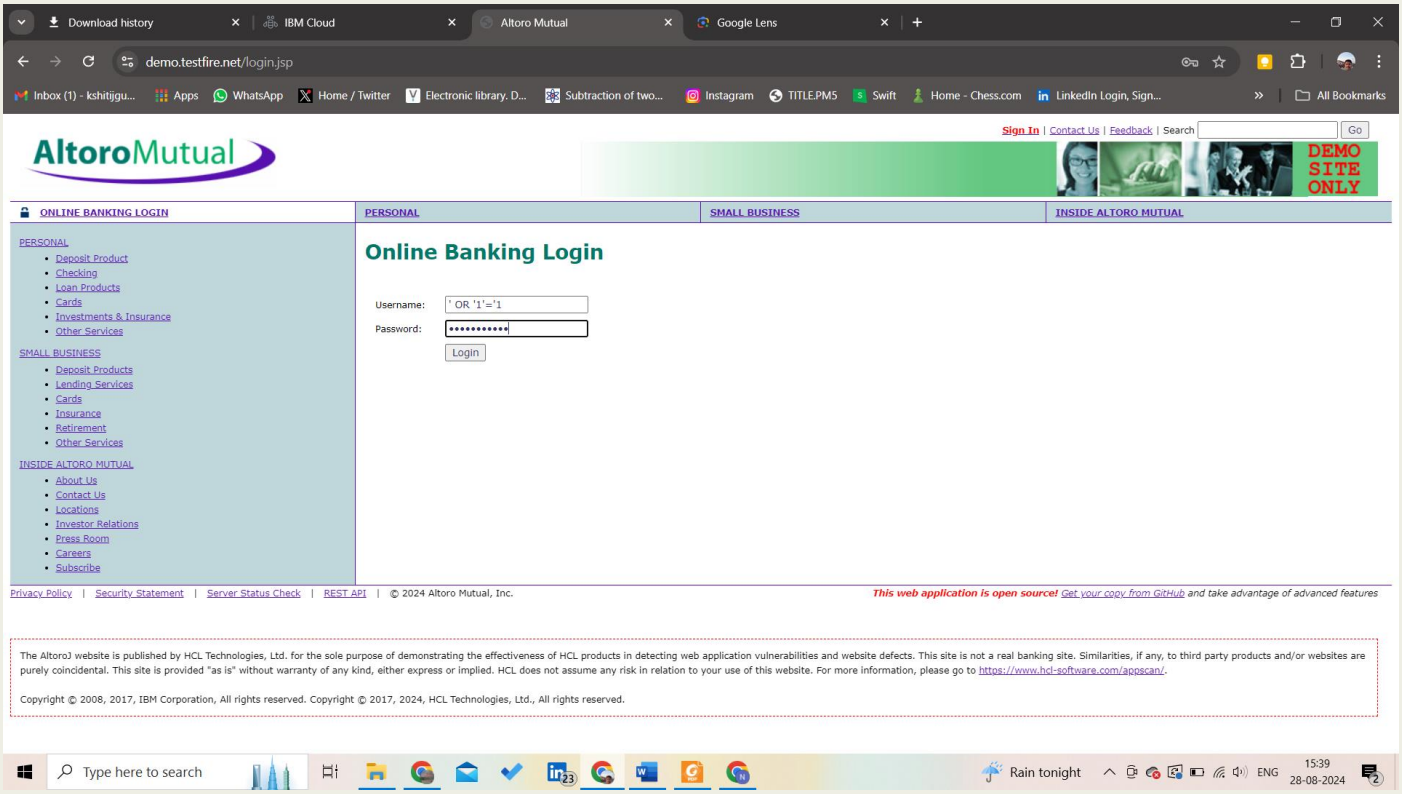


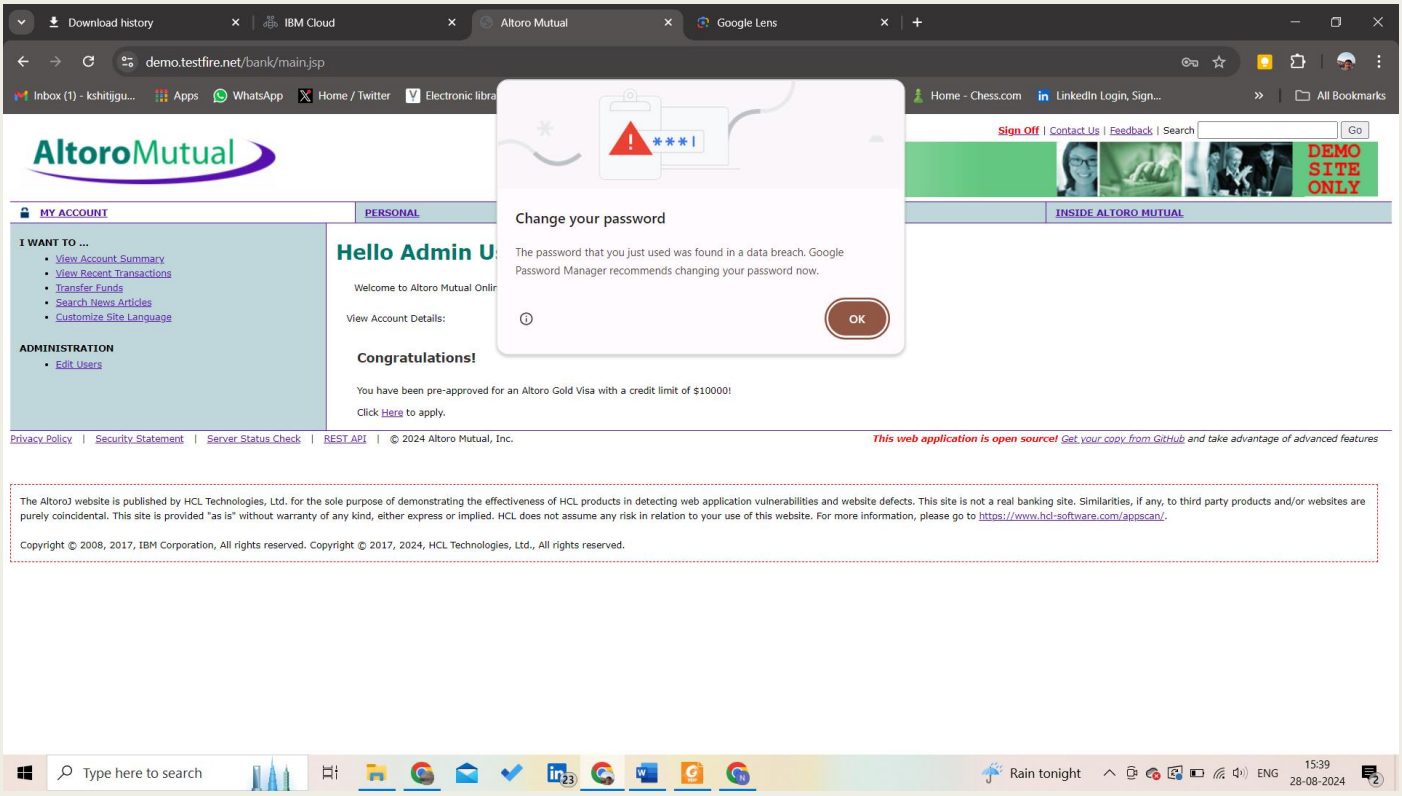
Now we will try to fetch username and password by adding uname and pass as one of the columns along with FROM users—



TASK: https://demo.testfire.net/index.jsp?content=personal_deposit.htm Identify any 3 web application vulnerabilities and website defects in the provided link.

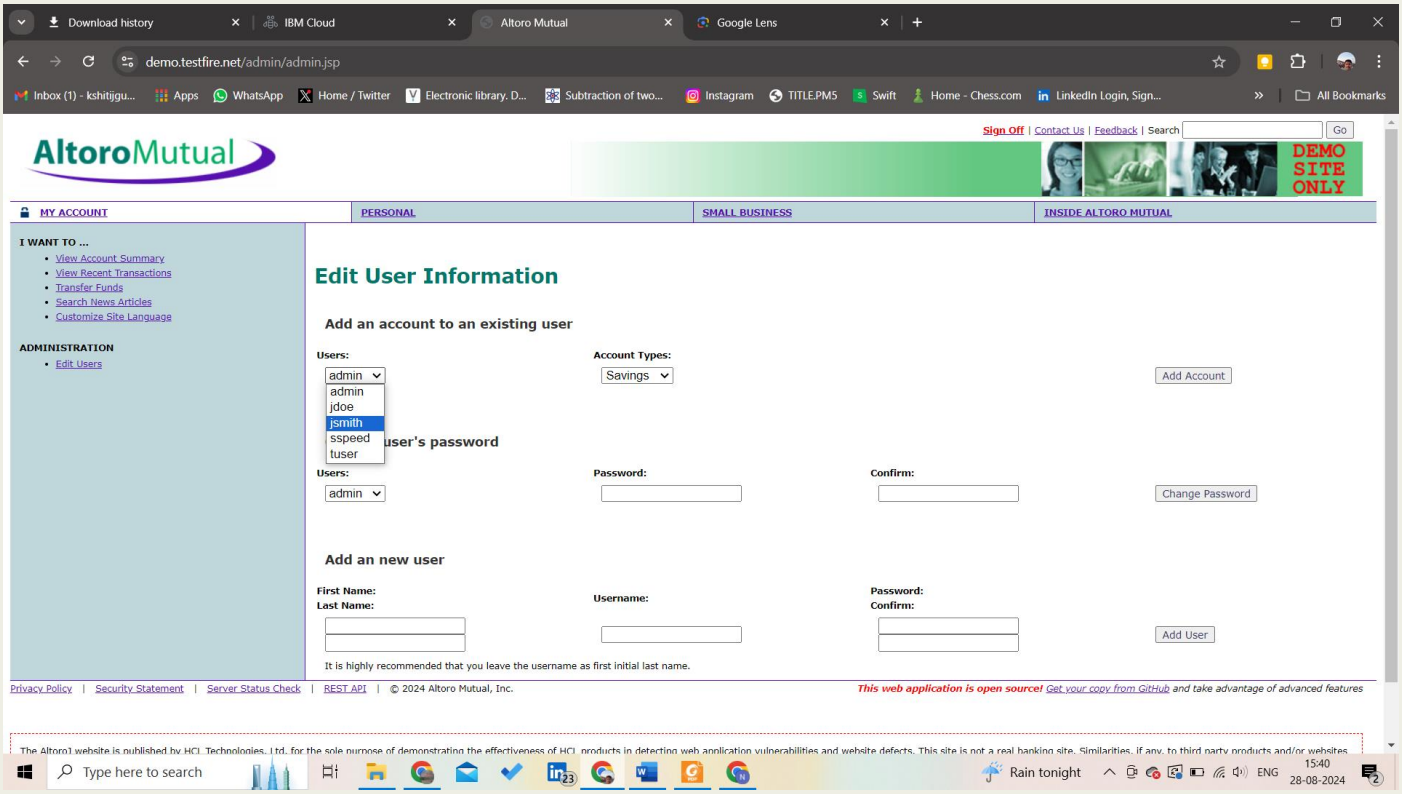
In the following link the first vulnerability is the SQL injection, where we add the ' OR '1'='1 as password and username to try to login.

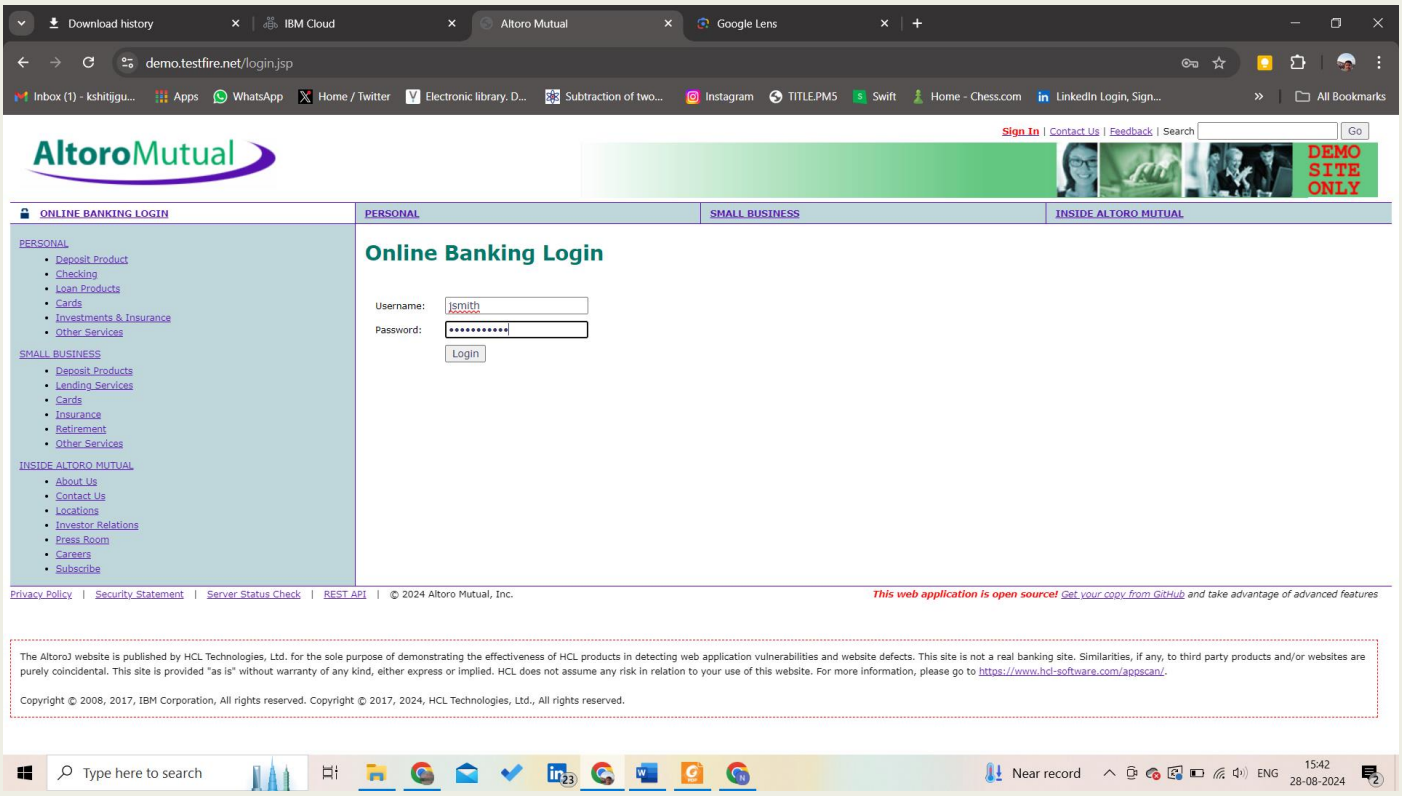


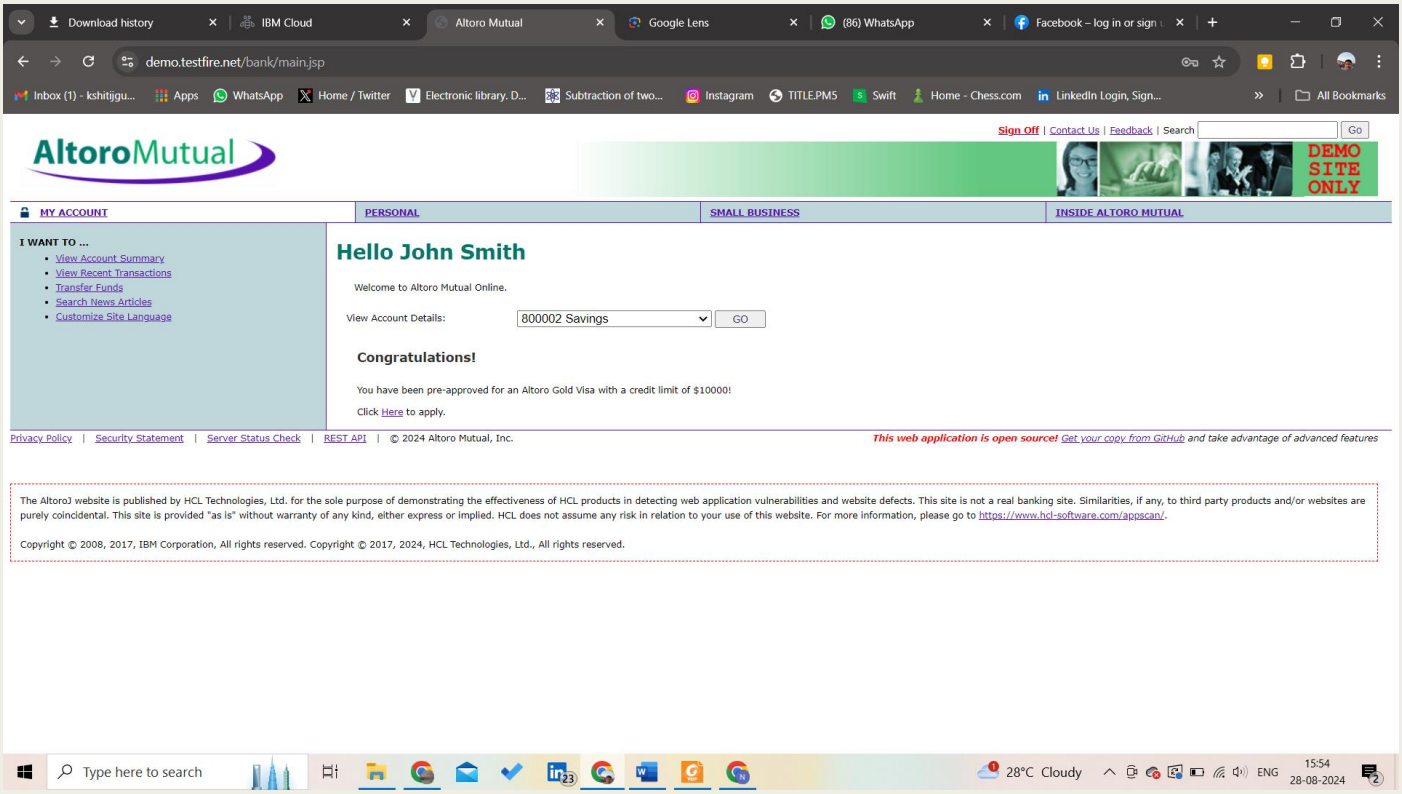


This injection attempts to bypass the login authentication by always evaluating the condition as true.

After doing this we can also find the list of users as we signed in as admin.







Now in the Search bar at the top right corner we can add <script> alert(document.cookie) </script> , here we can get the cookies details.