20155321 《网络攻防》 Exp8 Web基础

转载

weixin_30526593 于 2018-05-16 14:05:00 发布 42 收藏
文章标签： 数据库 php javascript ViewUI
原文链接：http://www.cnblogs.com/rafell/p/9045720.html
版权

20155321 《网络攻防》 Exp8 Web基础

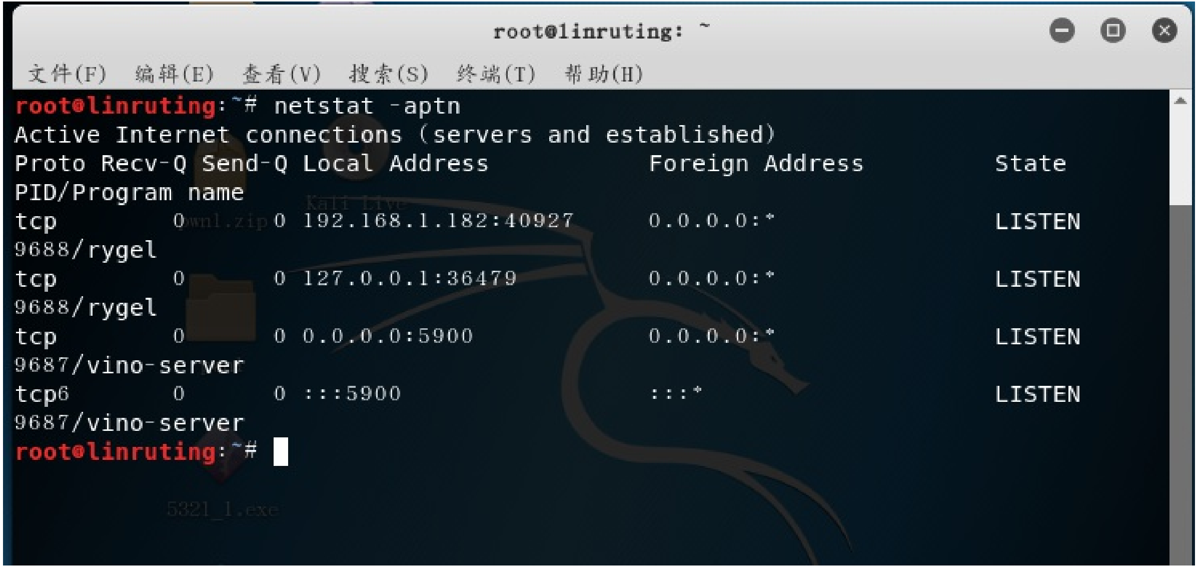
基础问题回答

- 什么是表单？
- 表单是主要负责数据采集功能。主要是以下三个部分构成：
  - 表单标签：包含处理表单数据所用的程序的URL以及数据提交到服务器的方法
  - 表单域：包含文本框、密码框、多行文本框、复选框、单选框、下拉选择框框等
  - 表单按钮：包含提交、复位和其他按，用于将数据传送到服务器上的或取消输入。
- 浏览器可以解析运行什么语言？
  - 浏览器肯定可以处理HTML/CSS，但对于JS脚本则可以调用JS脚本引擎进行处理
- WebServer支持哪些动态语言？
  - 最常用的三种动态网页语言ASP、JSP和PHP都可以被支持。

实验内容

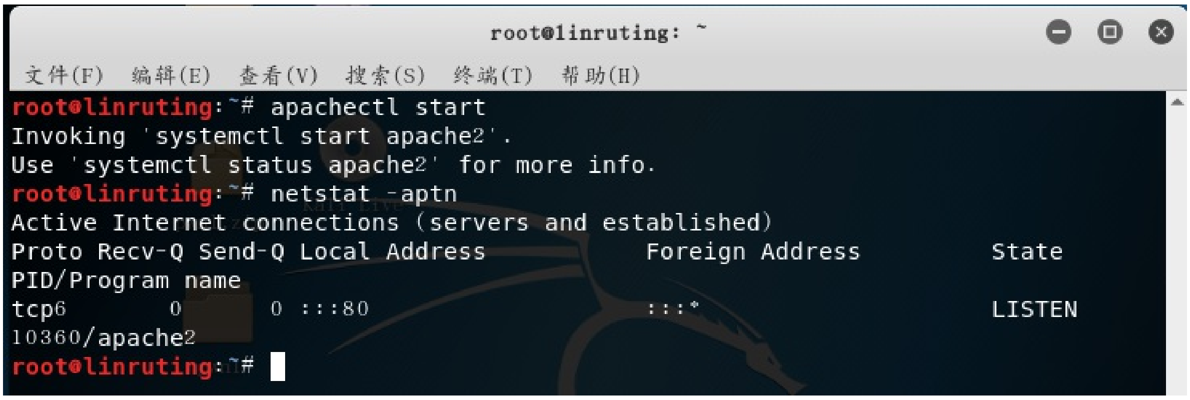
环境配置

- 使用`netstat -aptn`查看80端口是否被占用，发现我的电脑的80端口未被占用，也可以把这些用不到的进程先kill掉



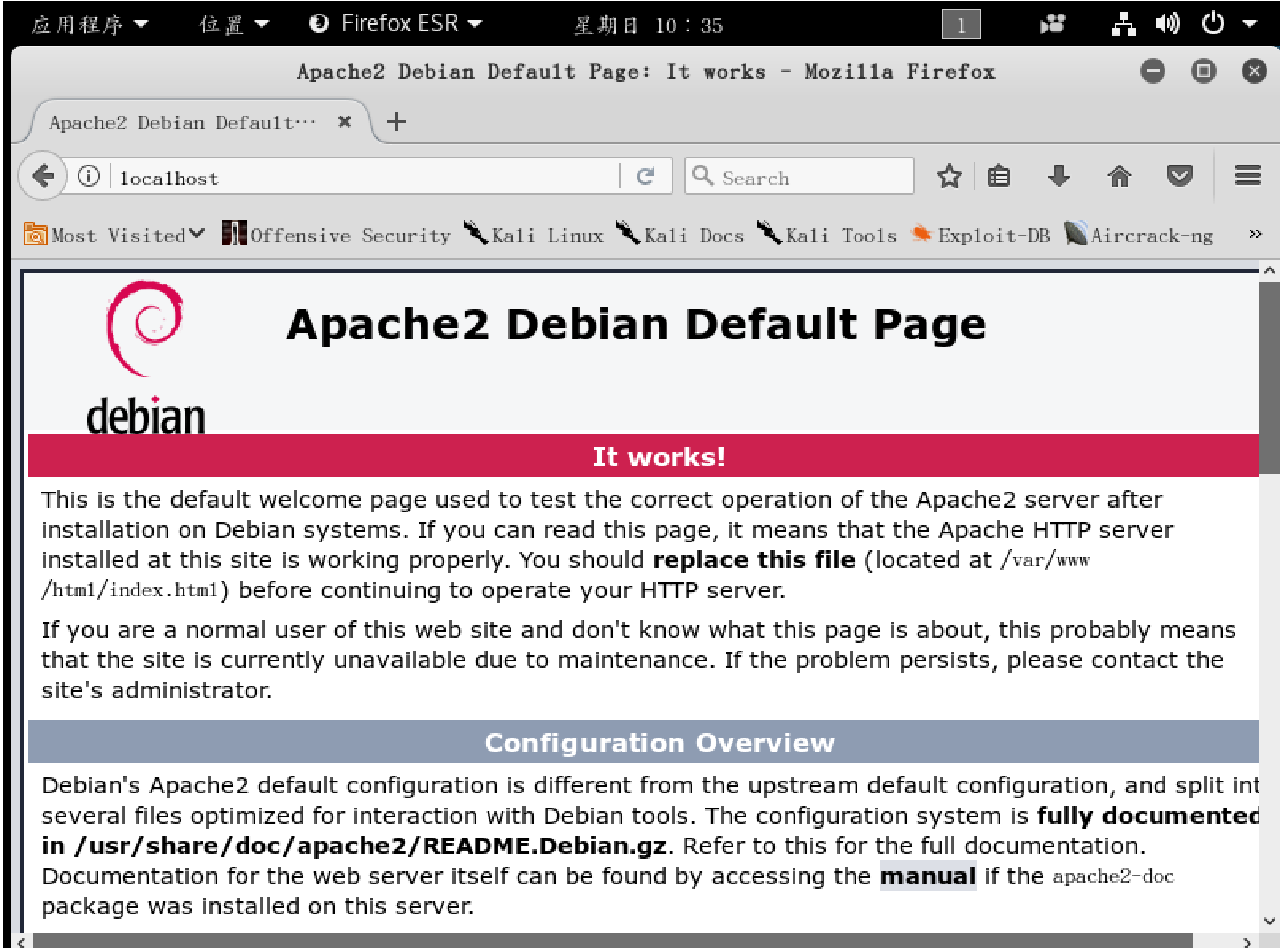
```
root@linruting:~# netstat -aptn
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0 192.168.1.182:40927     0.0.0.0:*               LISTEN
9688/rygel
tcp        0      0 127.0.0.1:36479         0.0.0.0:*               LISTEN
9688/rygel
tcp        0      0 0.0.0.0:5900            0.0.0.0:*               LISTEN
9687/vino-server
tcp6       0      0 :::5900                 :::*                    LISTEN
9687/vino-server
root@linruting:~# 
```

- 如果空闲就用`apachectl start`开启Apache，然后再次用`netstat -aptn`可以发现有80端口已被占用了



```
root@linruting:~# apachectl start
Invoking 'systemctl start apache2'.
Use 'systemctl status apache2' for more info.
root@linruting:~# netstat -aptn
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp6       0      0 :::80                   :::*                    LISTEN
10360/apache2
root@linruting:~#
```

- 打开浏览器，输入网址localhost查看apache是否可用，发现没有问题



Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

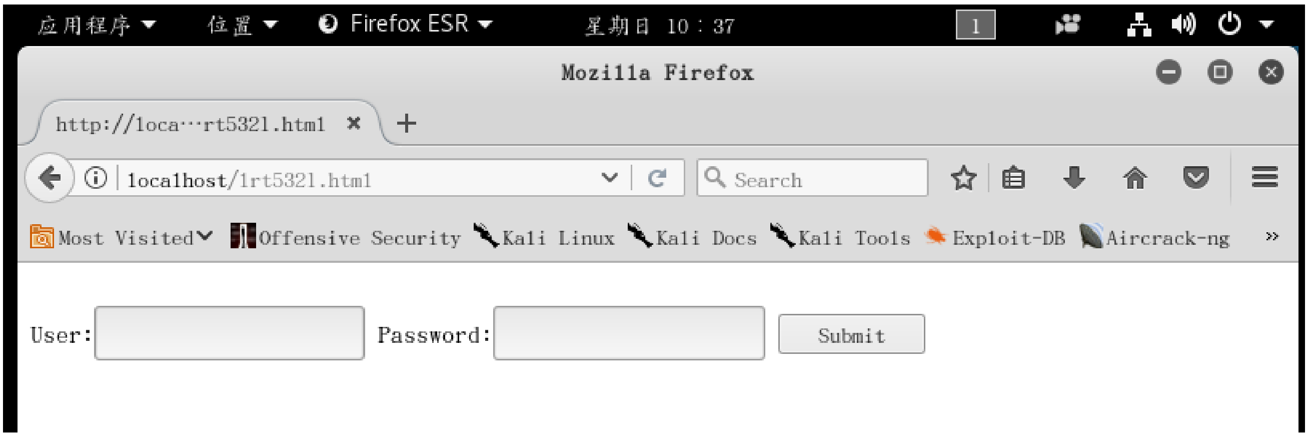
Debian's Apache2 default configuration is different from the upstream default configuration, and split int several files optimized for interaction with Debian tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

前端编程

- 使用cd /var/www/html在/var/www/html目录下编辑lrt5321.html

```
<form name="form" action="Login.php" method="post">
User:<input type="text" name="user">
Password:<input type="password" name="pw">
<input type="button" value="Submit" >
```

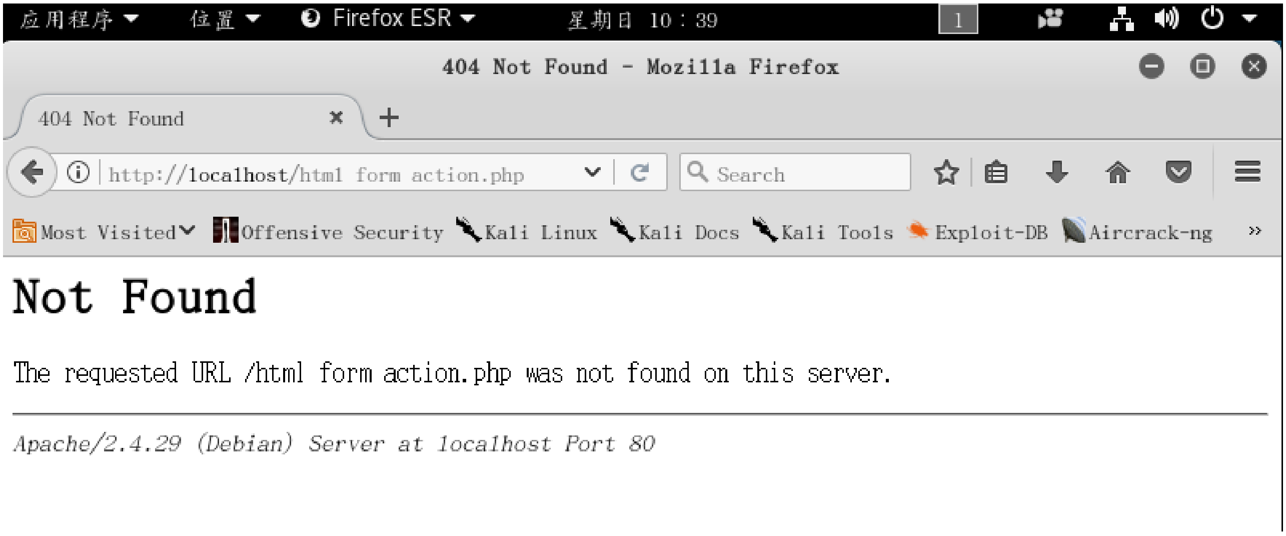
- 在firefox浏览器中输入网址localhost/lrt5321.html打开该网页


应用程序 ▾ 位置 ▾ Firefox ESR ▾ 星期日 10：37
Mozilla Firefox
http://loca…rt5321.html
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng
Submit

localhost/lrt5321.html

User: Password:

- 此时Submit也会跳转，但是跳转的页面是找不到的


应用程序 ▾ 位置 ▾ Firefox ESR ▾ 星期日 10：39
404 Not Found - Mozilla Firefox
404 Not Found
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng

http://localhost/html form action.php

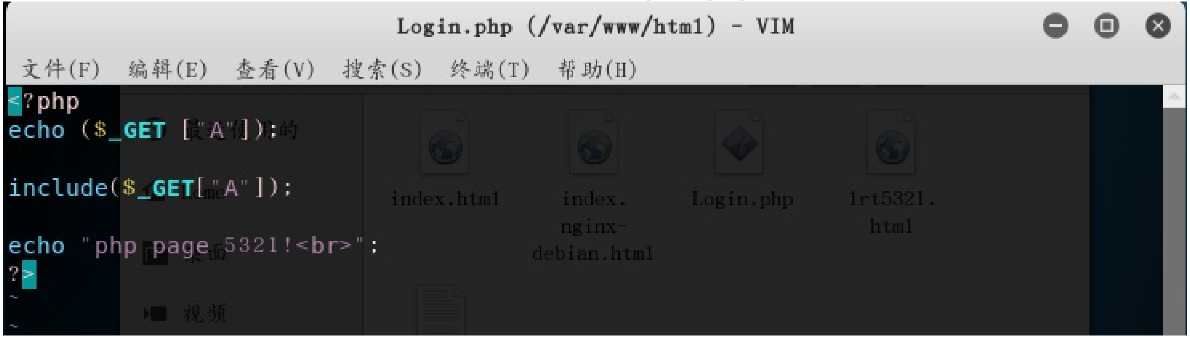
Not Found

The requested URL /html form action.php was not found on this server.

Apache/2.4.29 (Debian) Server at localhost Port 80

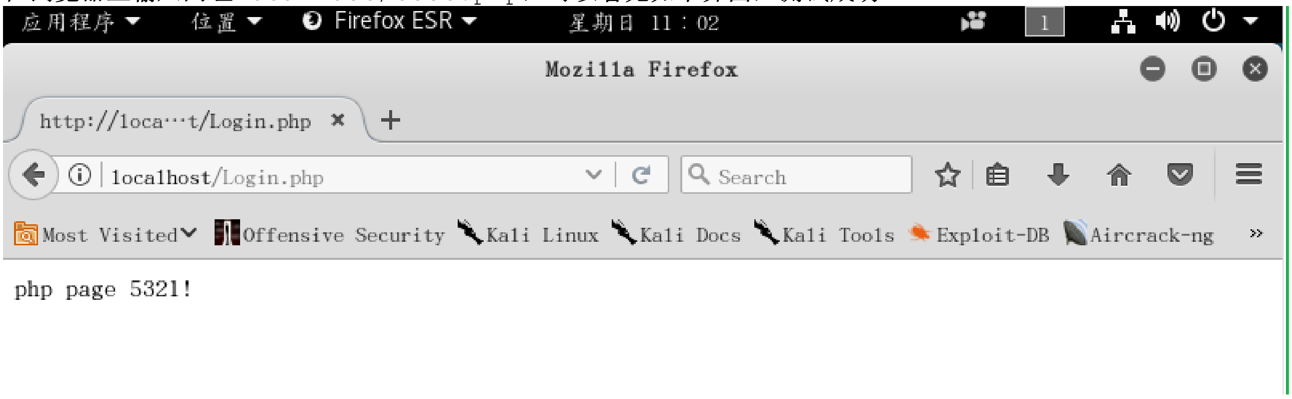
PHP测试

- 新建并编写一个PHP测试文件vi /var/www/html/Login.php


Login.php (/var/www/html) - VIM
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

```
<?php
echo ($_GET ["A"]);

include($_GET["A"]);

echo "php page 5321!<br>";
?>
~
~
```

- 在浏览器上输入网址localhost/test.php，可以看见如下界面，测试成功

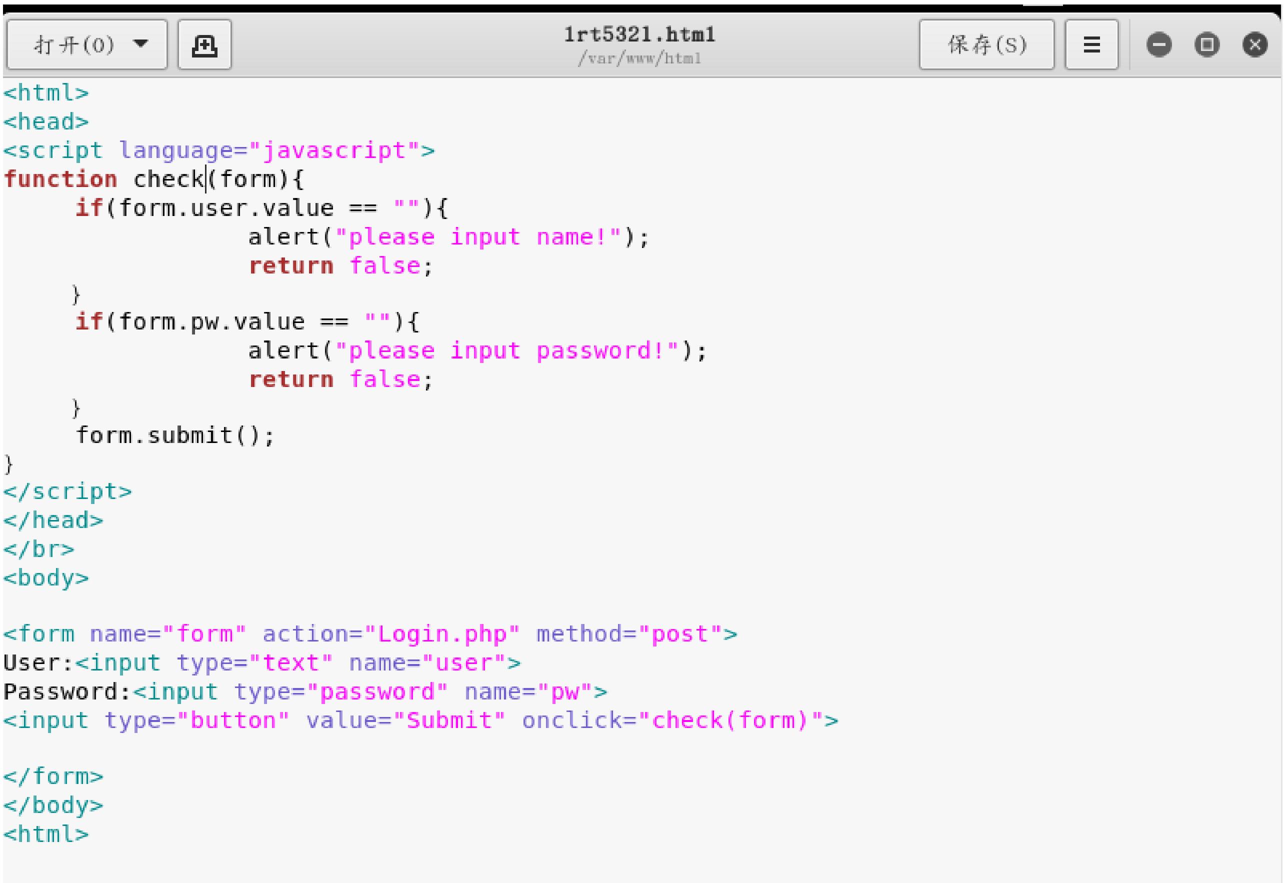

应用程序 ▾ 位置 ▾ Firefox ESR ▾ 星期日 11：02
Mozilla Firefox
http://loca…t/Login.php
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng

localhost/Login.php

php page 5321!

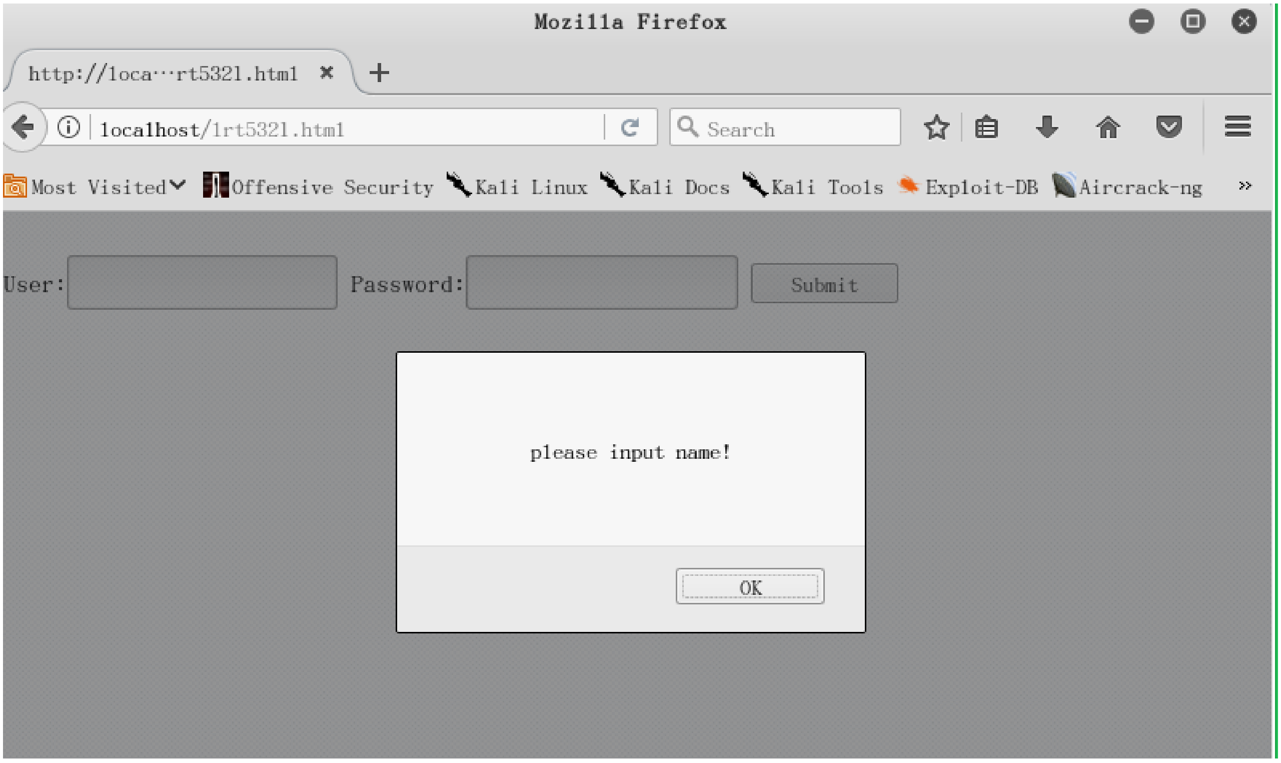
javascipt

- 使用javascipt修改之前lrt5321的代码了，完善Form表单，代码如下所示

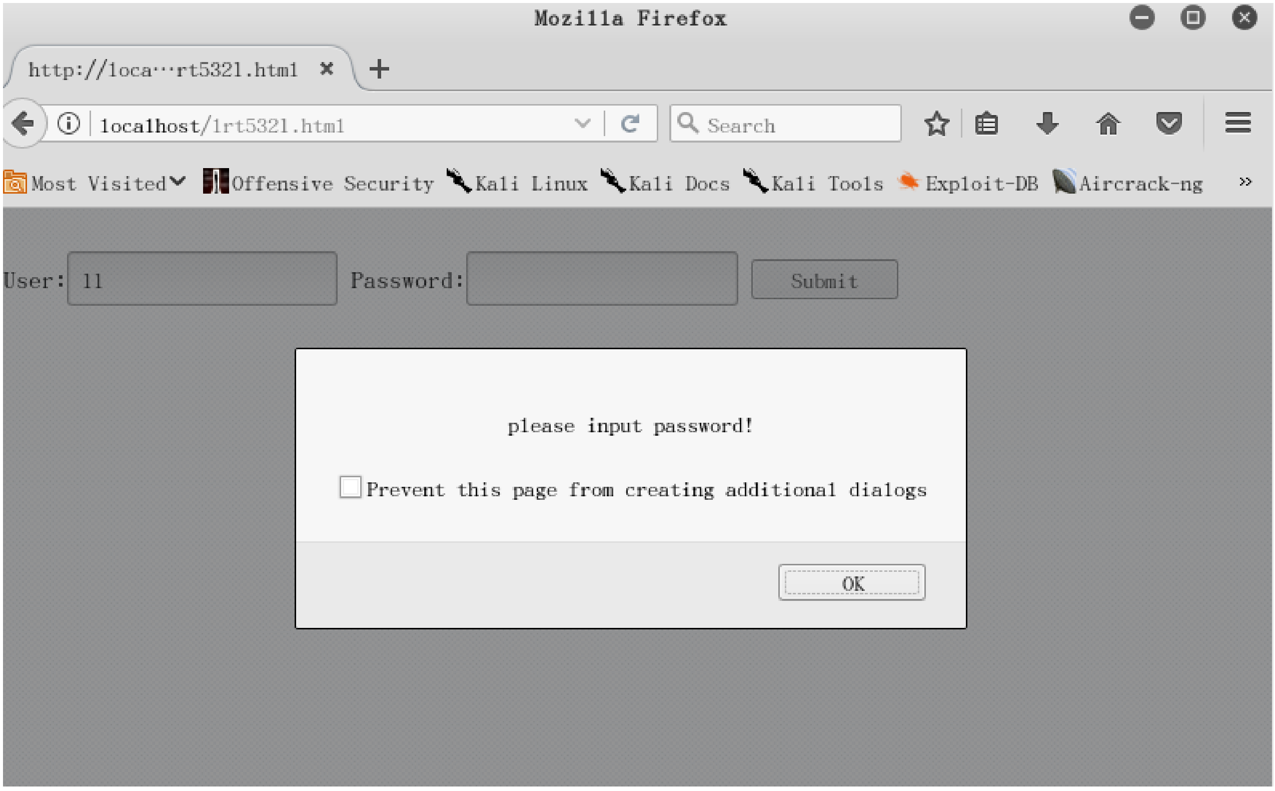


```
<html>
<head>
<script language="javascript">
function check(form){
     if(form.user.value == ""){
                 alert("please input name!");
                 return false;
     }
     if(form.pw.value == ""){
                 alert("please input password!");
                 return false;
     }
     form.submit();
}
</script>
</head>
</br>
<body>

<form name="form" action="Login.php" method="post">
User:<input type="text" name="user">
Password:<input type="password" name="pw">
<input type="button" value="Submit" onclick="check(form)">

</form>
</body>
<html>
```

- 在浏览器上测试，如下图所示



please input name!



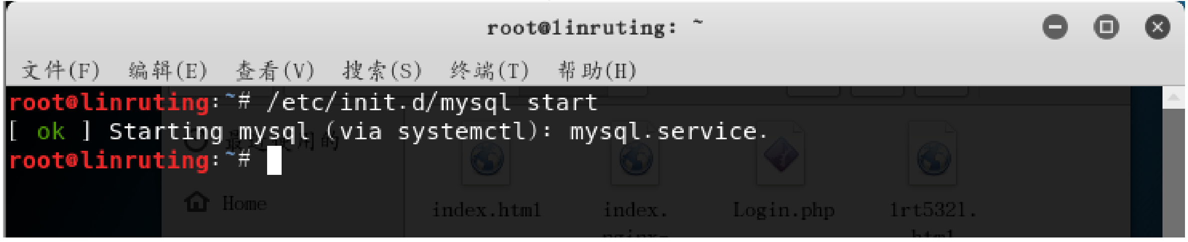
User: 11 Password:

please input password!

□ Prevent this page from creating additional dialogs

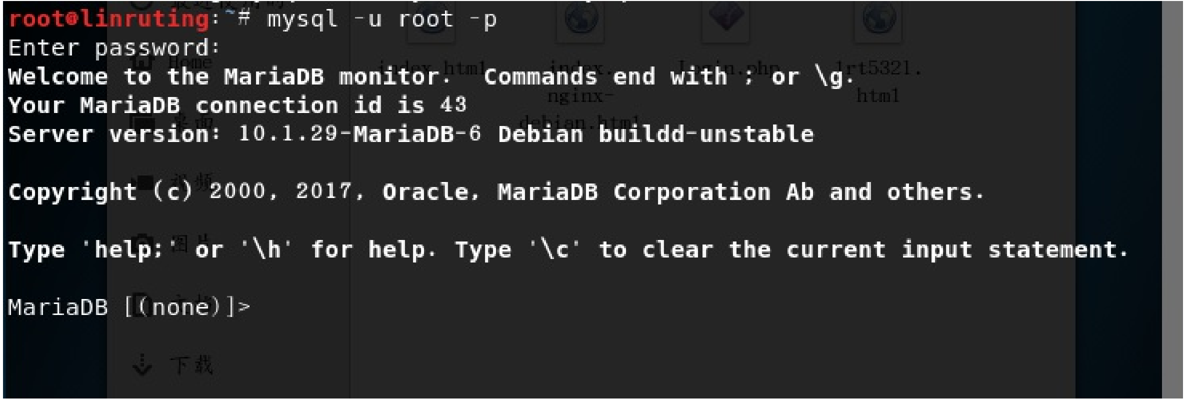
MySQL

- 输入命令`/etc/init.d/mysql start`开启sql服务



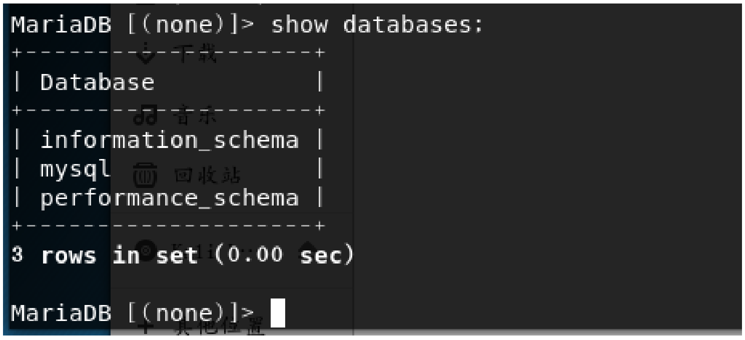
```
root@linruting:~# /etc/init.d/mysql start
[ ok ] Starting mysql (via systemctl): mysql.service.
root@linruting:~# 
```

- 输入命令`mysql -u root -p`用账号root登录



```
root@linruting:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 43
Server version: 10.1.29-MariaDB-6 Debian buildd-unstable

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

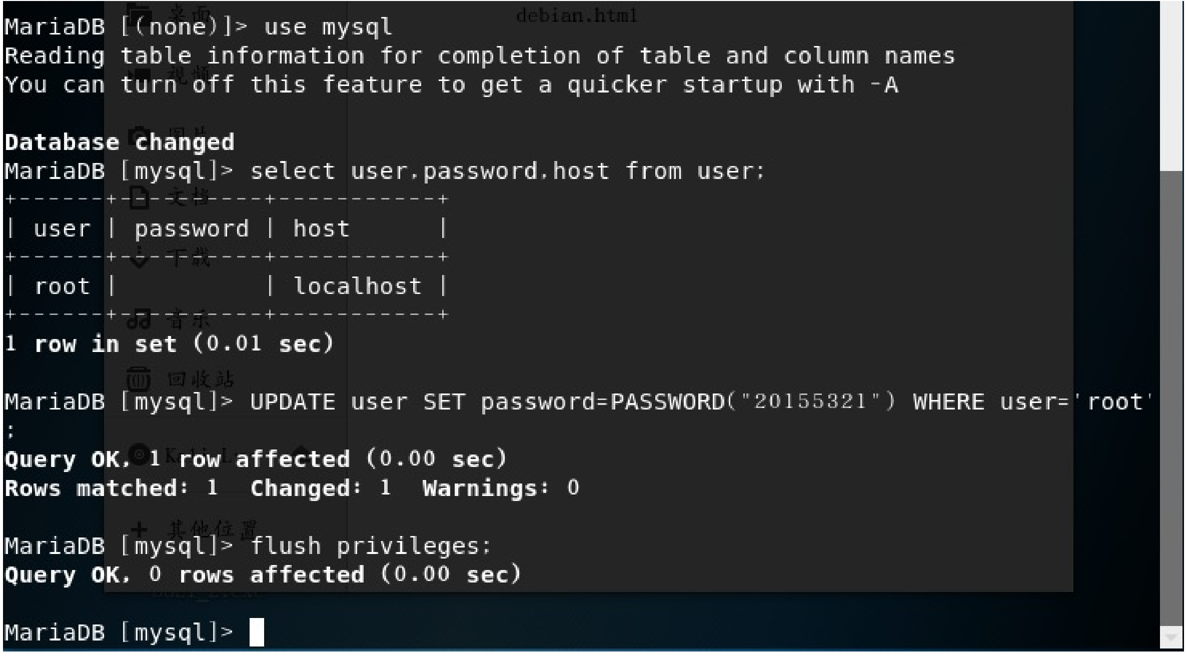
- 输入命令`show databases;`查看基本信息



```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
3 rows in set (0.00 sec)

MariaDB [(none)]> 
```

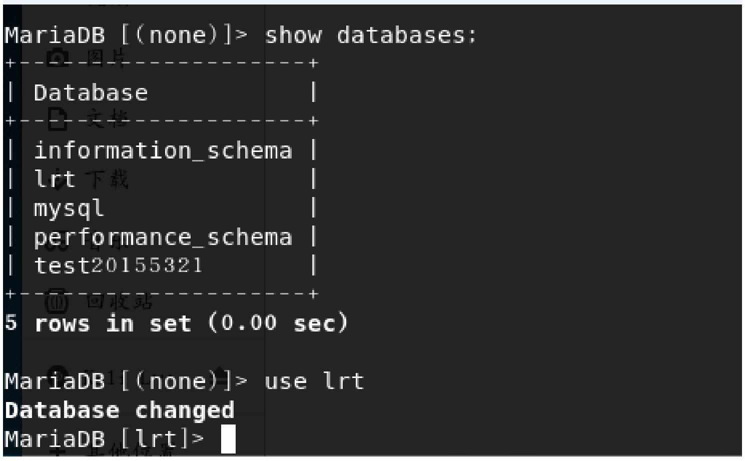
- 关于修改密码可以按照如下图所示进行
  - 用`use mysql;`，选择mysql数据库

- 用`select user, password, host from user;`，查看数据库的相关信息
- 输入`UPDATE user SET password=PASSWORD("新密码") WHERE user='root';`更改密码
- 用`flush privileges;`进行更新



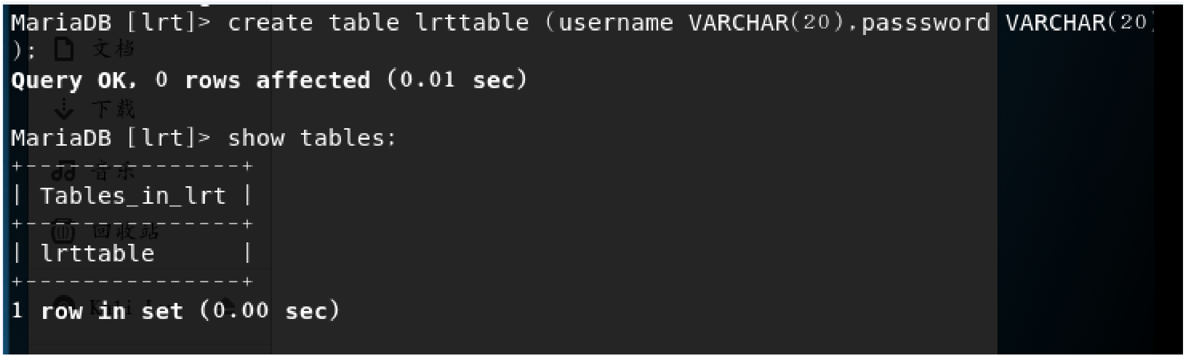
```
MariaDB [(none)]> use mysql
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [mysql]> select user,password,host from user;
+------+----------+-----------+
| user | password | host      |
+------+----------+-----------+
| root |          | localhost |
+------+----------+-----------+
1 row in set (0.01 sec)

MariaDB [mysql]> UPDATE user SET password=PASSWORD("20155321") WHERE user='root'
;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [mysql]> flush privileges;
Query OK, 0 rows affected (0.00 sec)

MariaDB [mysql]> 
```

- 输入命令`create database lrt` 建立一个数据库，输入命令`use lrt`使用这个数据库



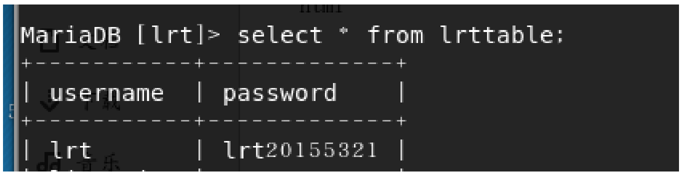
```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| lrt                |
| mysql              |
| performance_schema |
| test20155321       |
+--------------------+
5 rows in set (0.00 sec)

MariaDB [(none)]> use lrt
Database changed
MariaDB [lrt]> 
```

输入命令`create table lrttable (username VARCHAR(20),password VARCHAR(20));`建立一个数据表，再输入命令`show tables`查看当前的数据表



```
MariaDB [lrt]> create table lrttable (username VARCHAR(20),passsword VARCHAR(20
));
Query OK, 0 rows affected (0.01 sec)

MariaDB [lrt]> show tables;
+---------------+
| Tables_in_lrt |
+---------------+
| lrttable      |
+---------------+
1 row in set (0.00 sec)
```

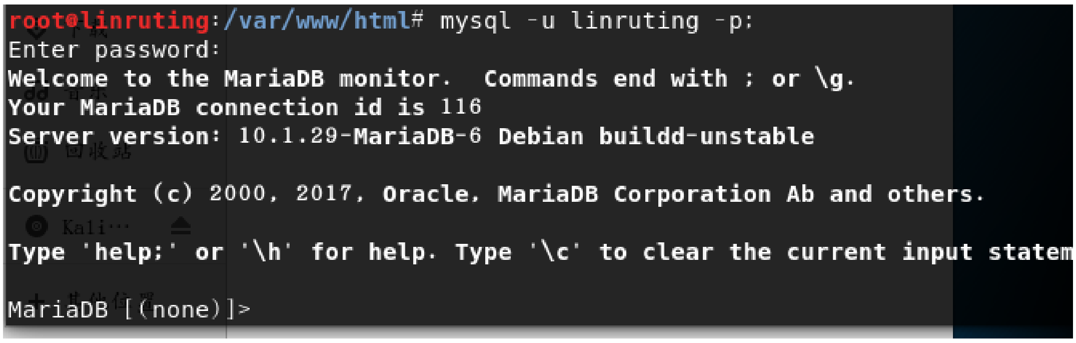
输入`insert into lrttable('lrt','lrt20155321');`添加数据库的信息



```
MariaDB [lrt]> select * from lrttable;
+----------+-------------+
| username | password    |
+----------+-------------+
| lrt      | lrt20155321 |
```

- 在MySQL中增加新用户，使用`grant select,insert,update,delete on 数据库.* to 用户名@localhost` identified by "密码";指令是将对某数据库的所有表的select,insert,update,delete权限授予某用户

之后用新的用户进行登录即可成功



```
root@linruting:/var/www/html# mysql -u linruting -p;
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 116
Server version: 10.1.29-MariaDB-6 Debian buildd-unstable

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statem

MariaDB [(none)]>
```

Web后端

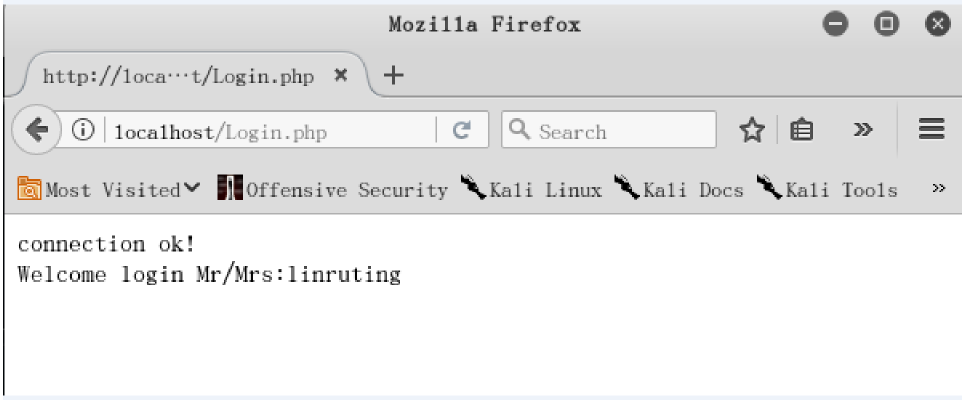
- 再次修改lrt5321.html文件，编写登录网页

```
<html>
<body>
<table>
    <form method ="POST" action="Login.php" name="Login"  >
    <tr>
    <td>user</td>
       <td><input type="text" name="username"  size="100" maxlength="100" onfocus="if (this.value=='Your na
    <td> </td>
    <td> </td>
    </tr>
    <tr>
    <td>password</td>
  <td><input type="password" name="password" size="100" maxlength="100" onfocus="if (this.value=='Your pass
    <td> </td>
    <td> </td>
    </tr>
    <tr>
    <td><input type="checkbox"  value="1">auto login</td>
    </tr>
    <table>
    <tr>
        <td><input type="submit"  value="Login" onClick="return validateLogin()"/></td>
            <td><input type="reset"  value="reset" /></td>
        </tr>
    </table>
    </form>
</table>

<script language="javascript">
    function validateLogin(){
        var sUserName = document.Login.username.value ;
        var sPassword = document.Login.password.value ;
        if ((sUserName =="") || (sUserName=="Your name")){
            alert("user name");
            return false ;
        }

        if ((sPassword =="") || (sPassword=="Your password")){
            alert("password!");
            return false ;
        }

    }
</script>
</body>
</html>
```

- 再修改PHP文件，如下

```
<?php
$uname=($_POST["username"]);
$pwd=($_POST["password"]);
$query_str="select * from lrttable where username='$uname' and password='$pwd';";
$mysqli = new mysqli("127.0.0.1", "linruting", "19970728", "lrtt");

if ($mysqli->connect_errno) {
    printf("Connect failed: %s\n", $mysqli->connect_error);
    exit();
}
echo "connection ok!";

if ($result = $mysqli->multi_query($query_str)) {
    if ($result->num_rows > 0 ){
        echo "<br> Welcome login Mr/Mrs:$uname <br> ";
    } else {
        echo "<br> login failed!!!! <br> " ;
    }
    $result->close();
}
$mysqli->close();
?>
```

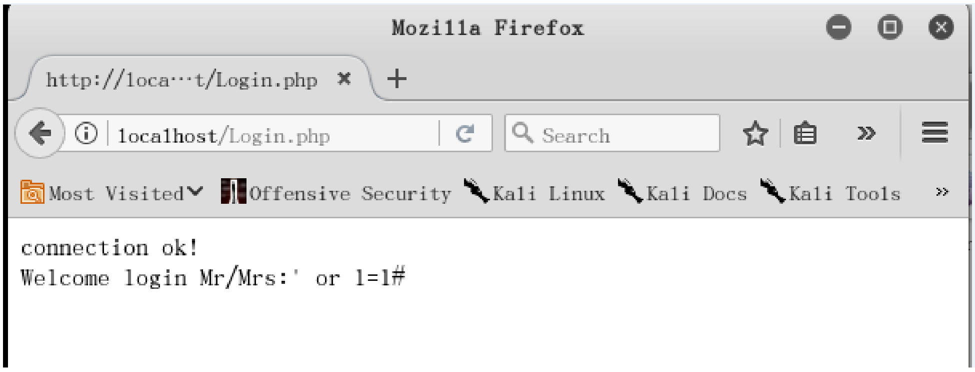
- 在浏览器输入`127.0.0.1/lrt5321.html`即访问自己的登录界面



```
connection ok!
Welcome login Mr/Mrs:linruting
```

SQL注入

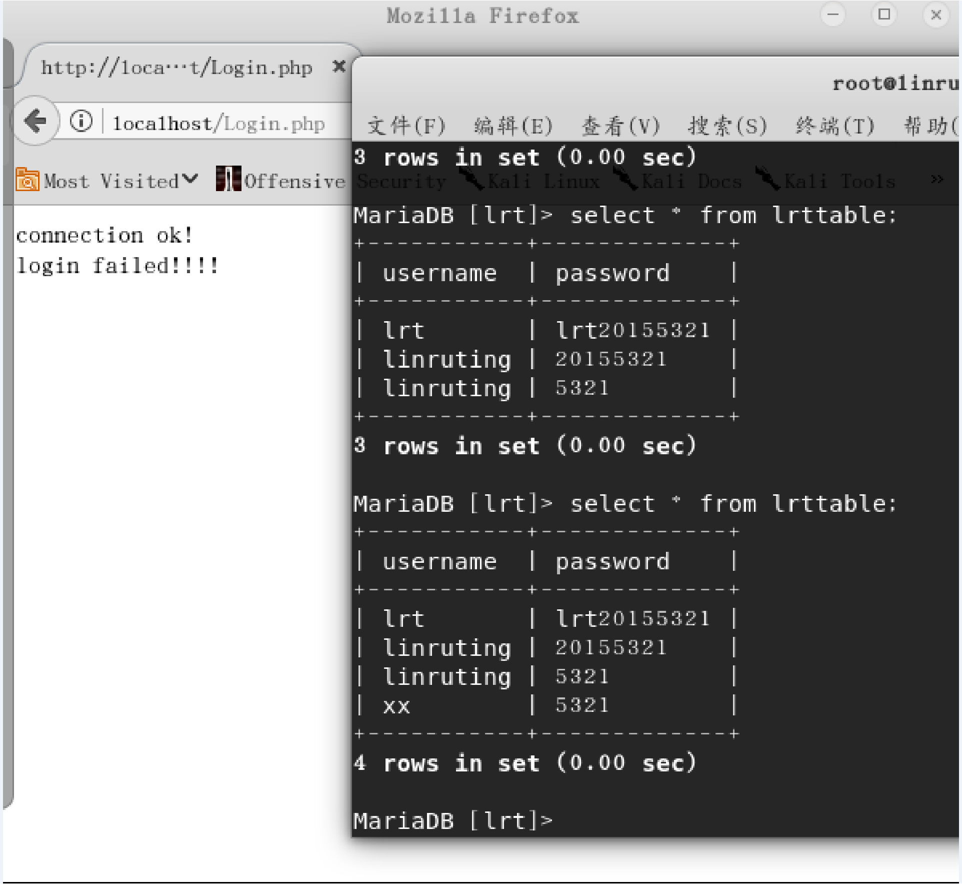
在用户名输入框中输入`' or 1=1#`，密码随便输入，这时候的合成后的SQL查询语句为`select * from lrttable where username='' or 1=1#' and password=''`



```
connection ok!
Welcome login Mr/Mrs:' or 1=1#
```

- 另外，还可通过SQL注入将用户名和密码保存在数据库中，但是要将`if ($result = $mysqli->query($query_str))`
  改成`if ($result = $mysqli->multi_query($query_str))`即实现执行多个sql语句，接着在用户名中输入`';insert into zxtable values('xx','53212');#`
  拆开来看就是`SELECT * FROM zxtable WHERE username='';`、`insert into zxtable`

values('xx','5321',);，接着登录，会显示登录失败，但是再去查数据库发现：



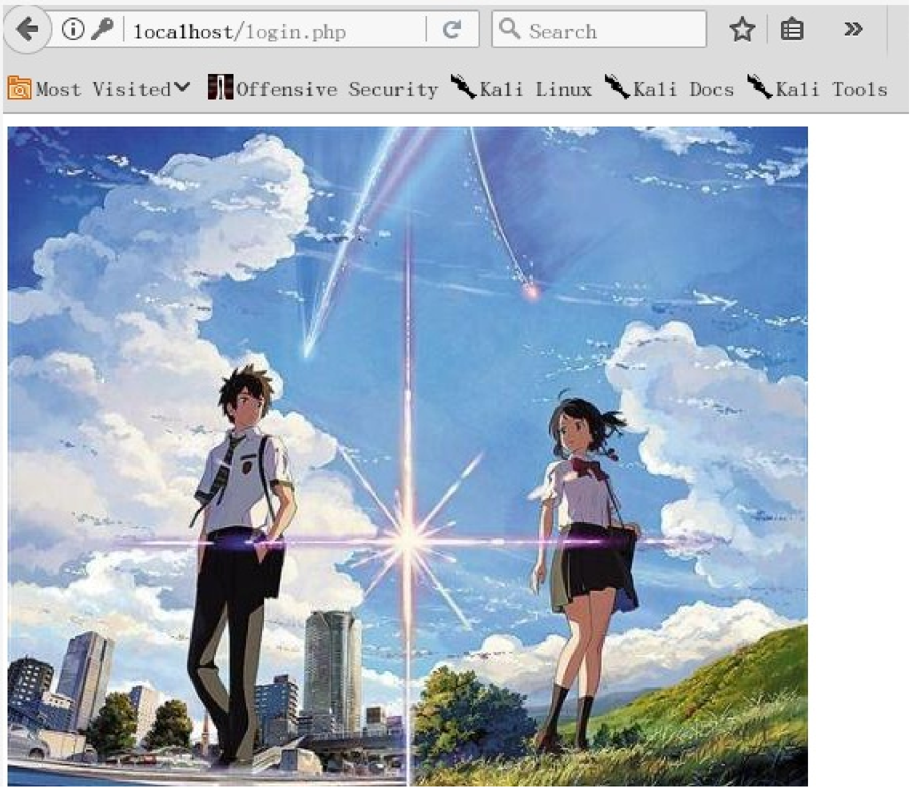
```
connection ok!
login failed!!!!
```

```
3 rows in set (0.00 sec)

MariaDB [lrt]> select * from lrttable;
+-----------+-------------+
| username  | password    |
+-----------+-------------+
| lrt       | lrt20155321 |
| linruting | 20155321    |
| linruting | 5321        |
+-----------+-------------+
3 rows in set (0.00 sec)

MariaDB [lrt]> select * from lrttable;
+-----------+-------------+
| username  | password    |
+-----------+-------------+
| lrt       | lrt20155321 |
| linruting | 20155321    |
| linruting | 5321        |
| xx        | 5321        |
+-----------+-------------+
4 rows in set (0.00 sec)

MariaDB [lrt]>
```

然后我们就可以再登录成功了。。。

XSS攻击

- XSS又叫跨站脚本攻击，它指的是恶意攻击者往Web页面里插入恶意html代码，当用户浏览该页之时，嵌入其中Web里面的html代码会被执行，从而达到恶意的特殊目的
- 在用户名中输入/var/www/html下的图片，再登录即可查看图片



实验总结与体会

- 总体而言，在有学长学姐的博客指导下，还是完成了本次实验。通过本次实验，让我重新复习了上学期网络安全编程的相关内容，虽然在实验的过程中，也因为代码的一些细节问题以及数据库中的一些操作被卡住了，但是最终还是通过自己不断地尝试解决了，在解决的过程中也对这方面的知识有了更深的理解，个人感觉还是有收获的。

转载于:https://www.cnblogs.com/rafell/p/9045720.html