

# CTF入门篇writeup——D0g3 Games

转载

[weixin\\_34029680](#) 于 2018-10-31 16:30:00 发布 459 收藏 1

文章标签: [php](#) [爬虫](#) [游戏](#)

原文链接: <https://yq.aliyun.com/articles/666668>

版权

今天在网上找到一个CTF的小游戏, 题目我做了几道感觉挺简单, 很适合入门, 之前了解CTF, 参加各种杯或者是看各种比赛题的writeup, 感觉太难了, 想到这我还是决定从点滴做起, 记录一下学习过程, 同时也想做一套CTF从入门到精通的教程。

网址: <http://ctf.d0g3.cn/>



image.png

下面就简单记录下每道题的解题过程, 希望能一点一点积累知识点

## WEB

### 1. \_GET

- 题目地址: <http://106.12.21.77:8080/get/get.php>

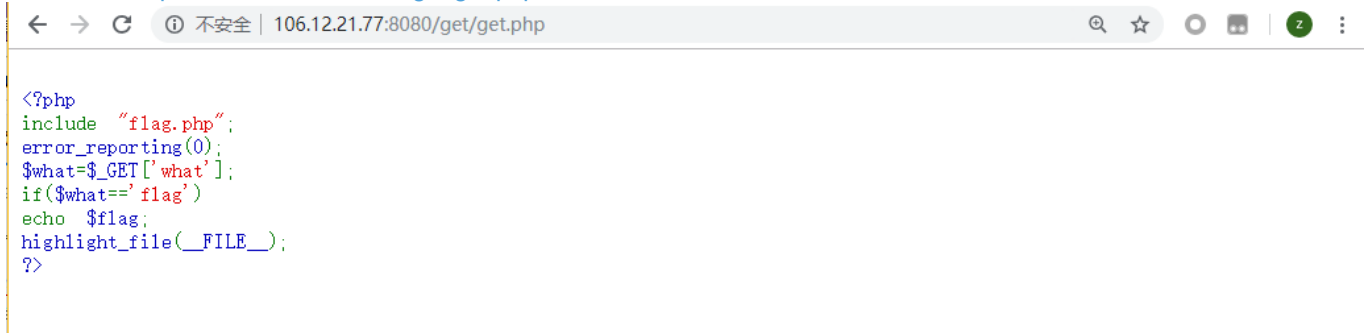


image.png

- 题目分析: 很简单了, 通过get接受一个变量what, 其值等于字符串flag.
- 解题方法:

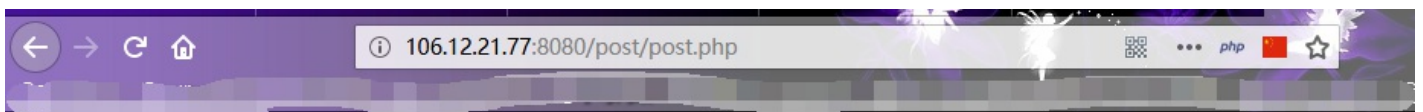
D0g3{get\_function}

```
<?php
include "flag.php";
error_reporting(0);
$what=$_GET['what'];
if($what=='flag')
echo $flag;
highlight_file(__FILE__);
?>
```

image.png

## 2. \_POST

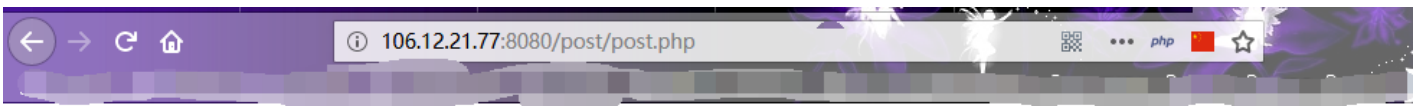
- 题目地址: <http://106.12.21.77:8080/post/post.php>



```
<?php
include "flag.php";
error_reporting(0);
$what=$_POST['what'];
if($what=='flag')
echo $flag;
highlight_file(__FILE__);
?>
```

image.png

- 题目分析: 同理同上, 只不过通过POST方式传参。
- 解题方法:



### D0g3{this\_is\_post\_function}

```
<?php
include "flag.php";
error_reporting(0);
$what=$_POST['what'];
if($what=='flag')
echo $flag;
highlight_file(__FILE__);
?>
```

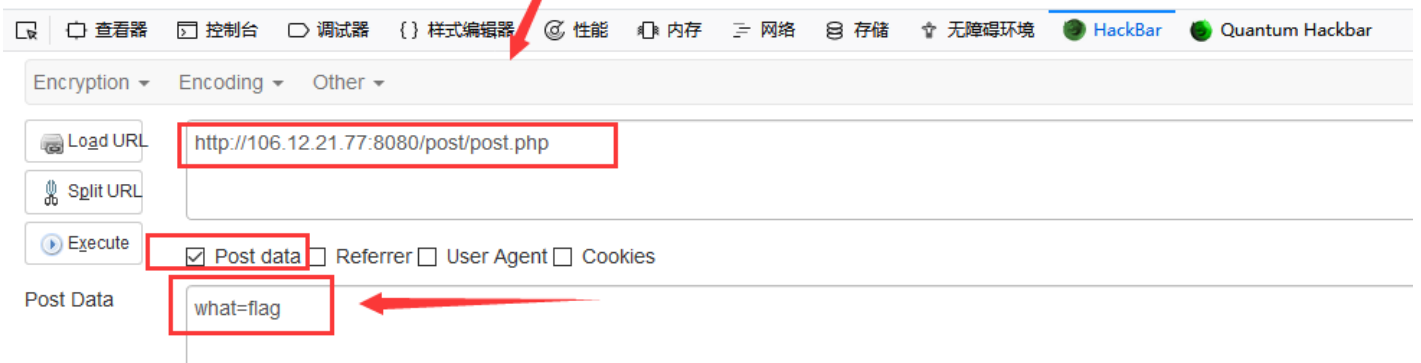


image.png

### 3. 突破物理极限

- 题目地址: <http://106.12.21.77:8080/length/length.html>



image.png

- 题目分析: 根据提示输入12345提交,但是由于限制只能输入123,因此一看就知道要绕过限制
- 解题方法: 方法不唯一可以抓包修改,也可以F12修改前端代码。

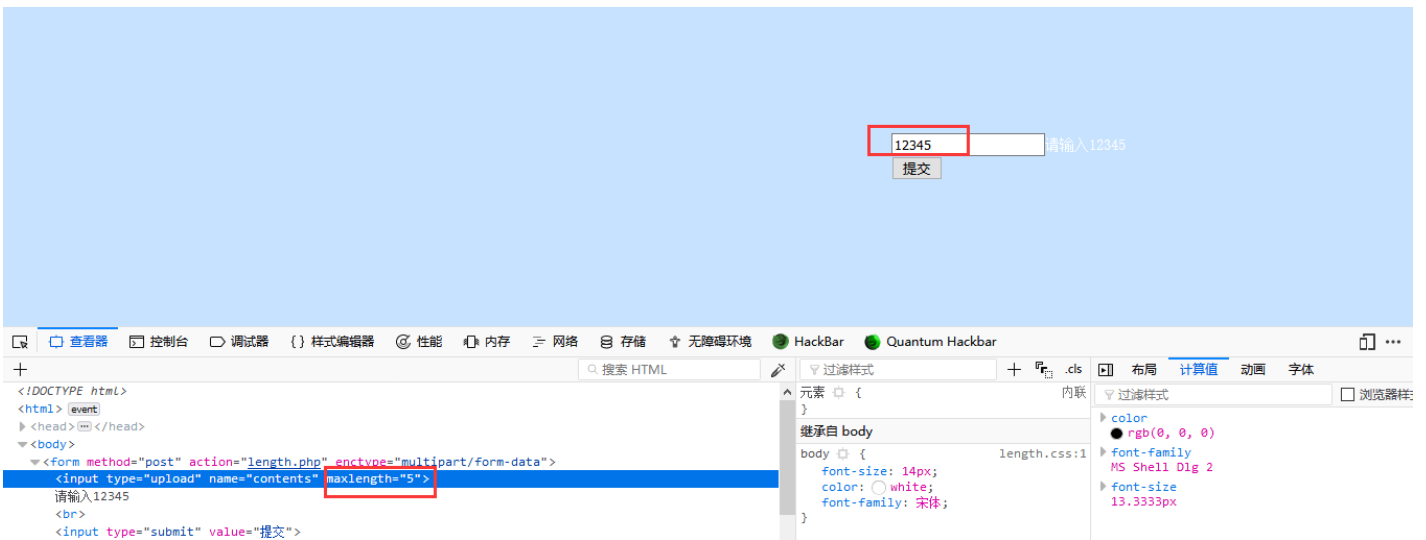
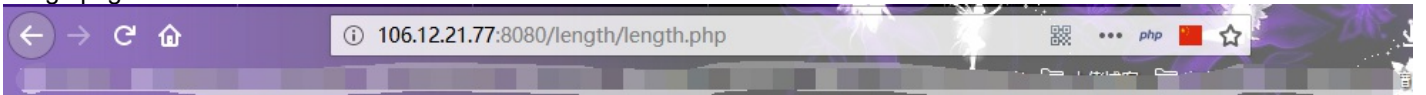


image.png

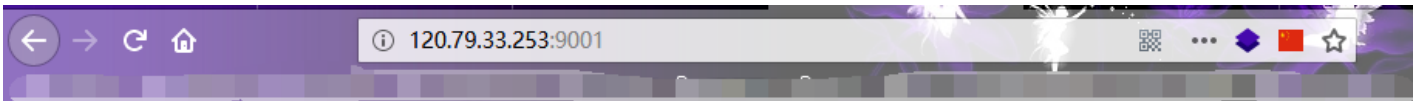


D0g3(max\_lenth)

image.png

#### 4. serialize

- 题目地址: <http://120.79.33.253:9001/>



```
<?php
error_reporting(0);
include "flag.php";
$KEY = "D0g3!!!";
$str = $_GET['str'];
if (unserialize($str) === "$KEY")
{
    echo "$flag";
}
show_source(__FILE__);
```

image.png

- 题目分析: 题目还是很简单的, 通过GET传参字符串str, 将其传入的字符串通过unserialize()反序列化, 使其反序列化结果值等于\$KEY, 于是解题方法就是传入事先通过serialize()函数序列化后的字符串。
- 解题过程:  
先写个简单序列化字符串的脚本

```
<?php
$key = $_GET['key'];
$str = serialize($key);
echo "$str";
?>
```

然后运行脚本，可以得到反序列化字符串脚本

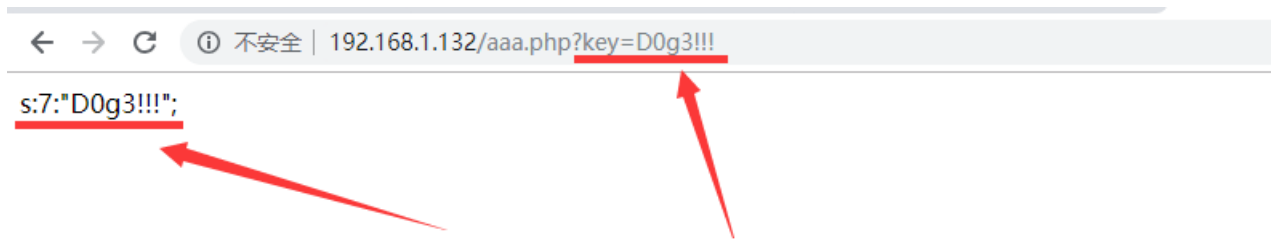


image.png  
最后我们解题



image.png

## 5. xss1

- 题目地址: <http://206.189.214.99:4080/xss/>



image.png



# d0g3 xss

没有找到和相关的结果.

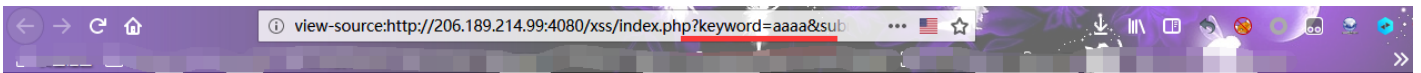


记得闭合标签,chrome内核浏览器(国内的qq, 360等浏览器用的也是chrome内核)有拦截, 请用其他浏览器打开

payload的长度:0

image.png

- 题目分析: 这道题是考察xss漏洞的, 根据提示弹窗即可, 想必也是讲xsspayload作为参数值传入, 那就随便输入值提交后先查看看页面源代码吧。



```

20 document.onmousedown = click;
21 document.oncontextmenu = new Function("return false;")
22 document.onkeydown = document.onkeyup = document.onkeypress = function () {
23     if (window.event.keyCode == 123||window.event.keyCode == 17) {
24         confirm("不让你用F12和Ctrl!!!");
25         window.event.returnValue = false;
26         return (false);
27     }
28 }
29 // <--123--112是F1-F12的代码数-->
30 window.alert = function()
31 {
32     confirm("恭喜你成功弹窗, 请联系qq索要flag, 喂喂喂: FDrag0n: 1021872752, passer6y: 749914034");
33 }
34 </script>
35 <title>欢迎来到d0g3 xss</title>
36 </head>
37 <body>
38 <h1 align=center>d0g3 xss</h1>
39 <h2 align=center>没有找到和aaaa相关的结果.</h2><center>
40 <form action=index.php method=GET>
41 <input name=keyword value="aaaa">
42 <input type=submit name=submit value="搜索"/>
43 </form>
44 </center><center><img src=logo.jpg></center>
45 <center>记得闭合标签,chrome内核浏览器(国内的qq, 360等浏览器用的也是chrome内核)有拦截, 请用其他浏览器打开</center>
46 <h3 align=center>payload的长度:4</h3></body>
47 </html>
48
49
50

```



image.png

- 解题方法: 根据题目分析, 构造payload传入到参数, 很简单了。通过测试发现两个输出点, , 第一个输出点实体编码了, 因此根据第二个输出点构造payload。

```
payload: "><script>alert(1)</script>
```

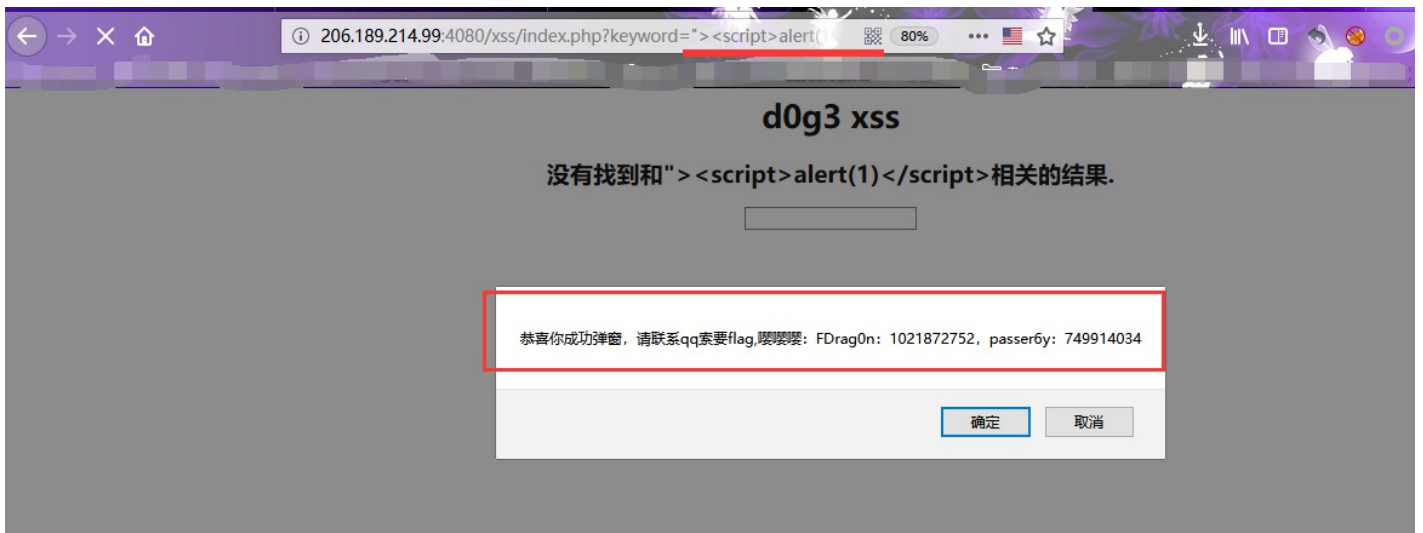


image.png

哇，弹出俩QQ，加好友索要flag，666，怎么不弄个公众号自动获取呢？

有趣的聊天截图，哈哈

披着羊皮的狼 10:51:56

flag呢

Passer6y 10:52:02

payload呢?

披着羊皮的狼 10:52:26

· "><script>alert(1)</scrip



Passer6y 10:53:32

d0g3{xss\_is\_easy}

披着羊皮的狼 10:53:53

## 6. htmlspecialchars

- 题目地址: <http://120.79.33.253:9004/?id=111>

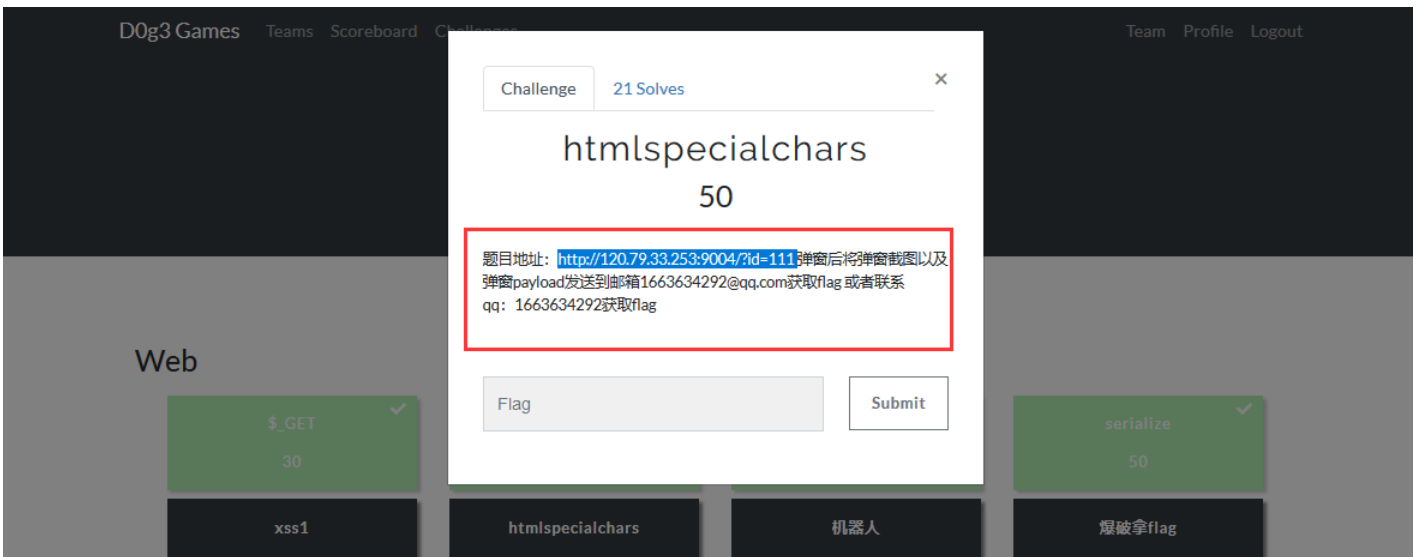


image.png

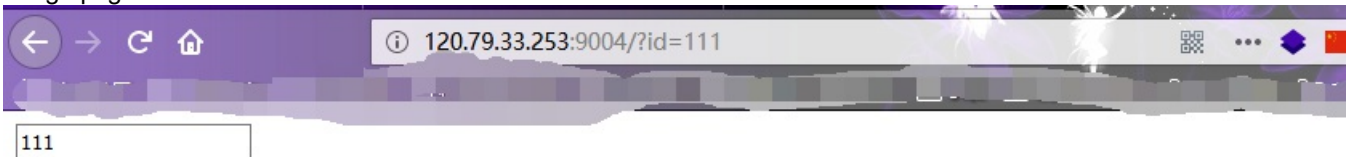


image.png

- 题目分析：哎，真麻烦，目标弹窗加qq，通过测试发现，后台使用htmlspecialchars对<>进行了HTML实体编码
- 解题方法：标签内构造XSSpayload

```
payload: ' onmouseover=alert(1)//
```

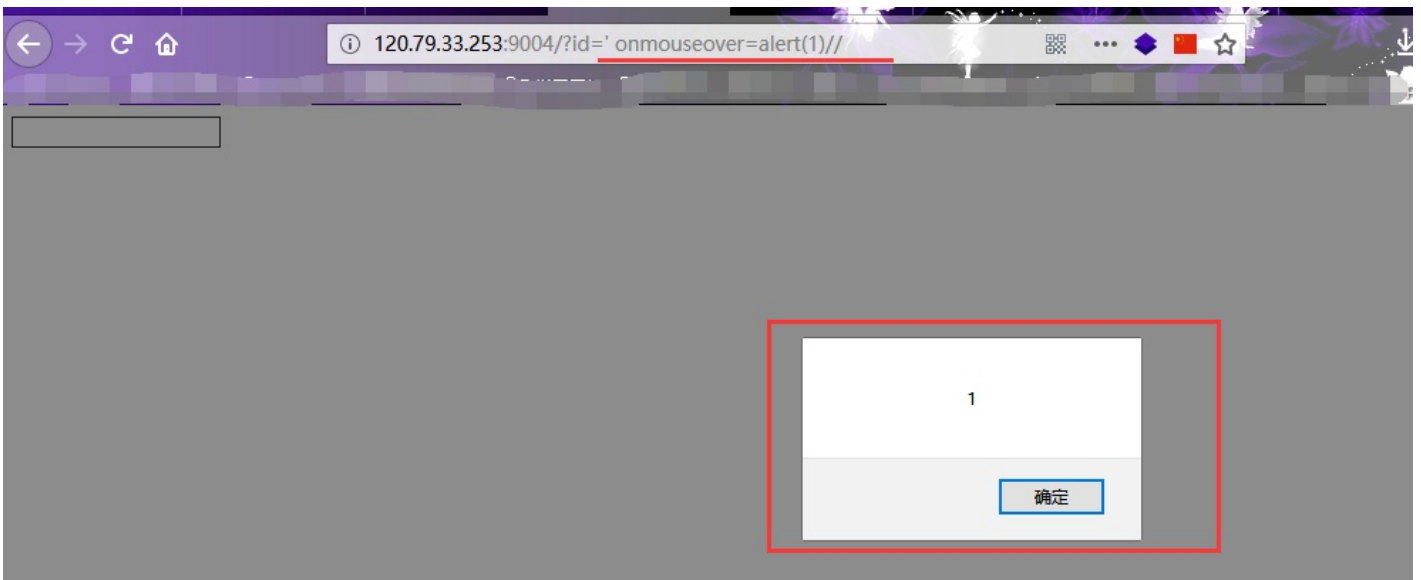
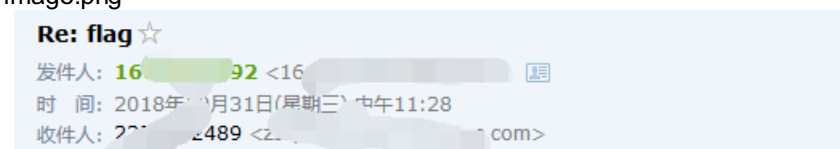


image.png



```
D0g3{7a915da604c1b26d03e0b494ffbcc3a9}
```

image.png



## 7. 机器人

- 题目地址: <http://106.12.21.77:8080/robots/robots.html>

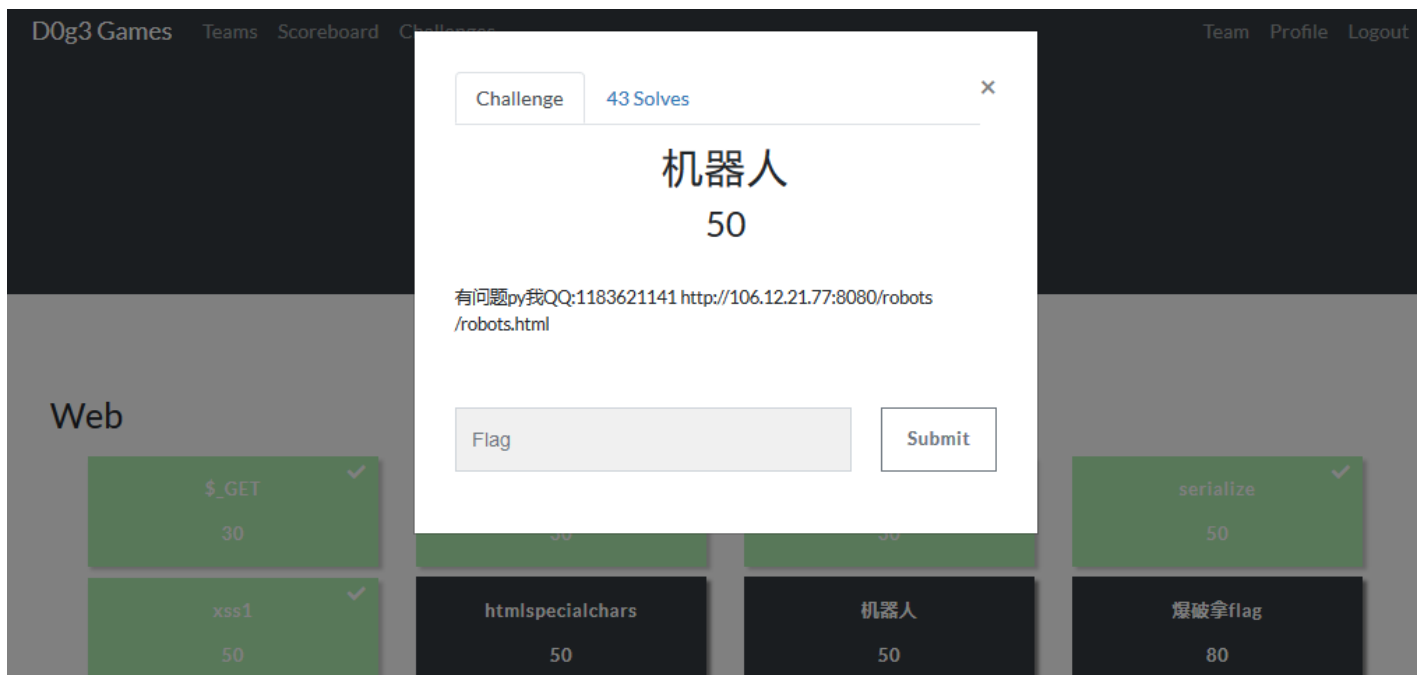


image.png

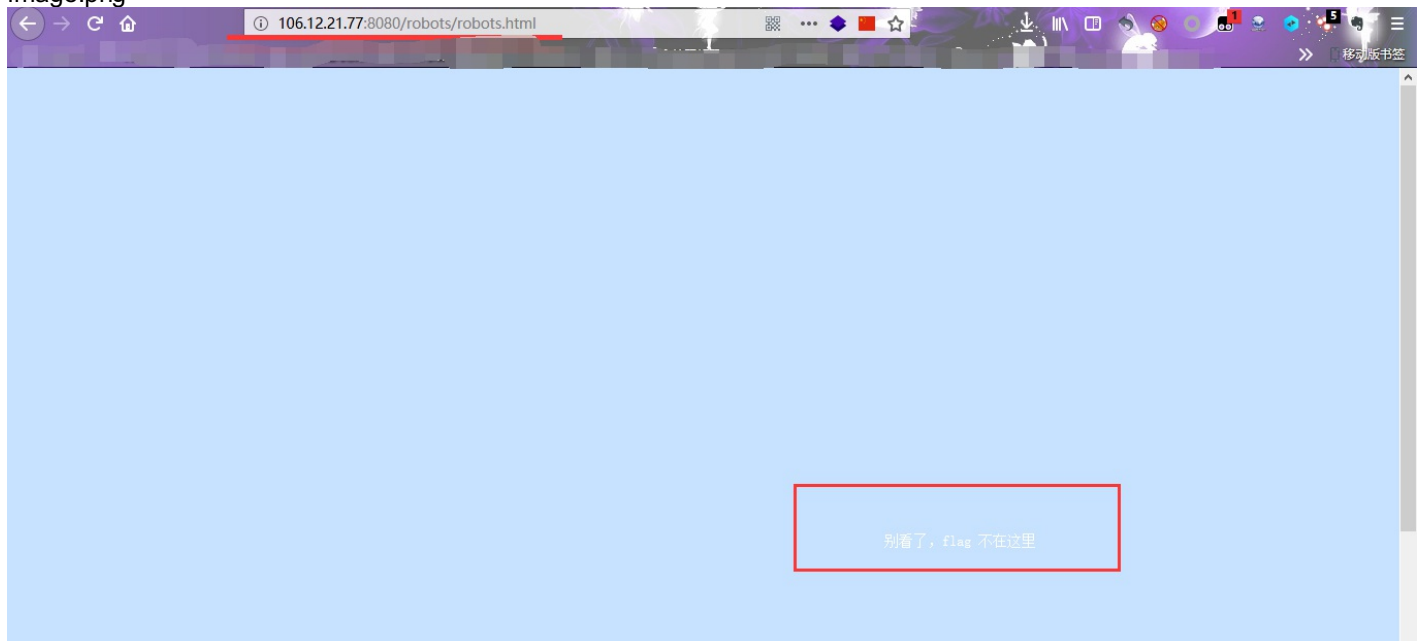
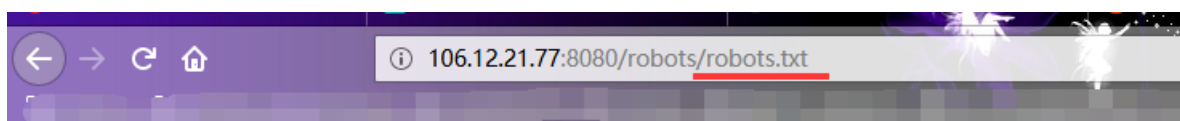


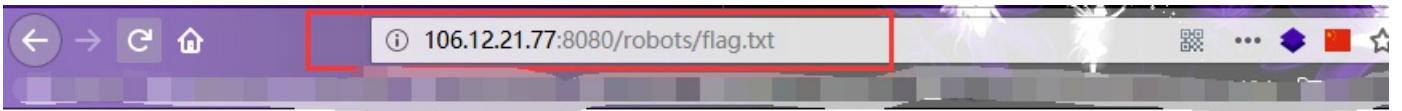
image.png

- 题目分析: 这道题凭直觉考察的是robots.txt, 简单解释一下就是, 为了防止搜索引擎的爬虫, 通常在网站目录下会有一个robots.txt文件, 来告诉搜索引擎哪些目录允许爬虫。那就试试呗。
- 解题方法: 访问robots.txt文件。果然有收获, 发现一个flag.txt的文件, 然后顺利访问拿到flag。



```
#  
# robots.txt for www.d0g3.cn  
#  
User-agent: *  
Disallow: /robots/flag.txt
```

image.png



D0g3 {robots\_txt\_is\_cool}

image.png

## 8. 爆破拿flag

- 题目地址: <http://106.12.21.77:8080/burp/burp.html>

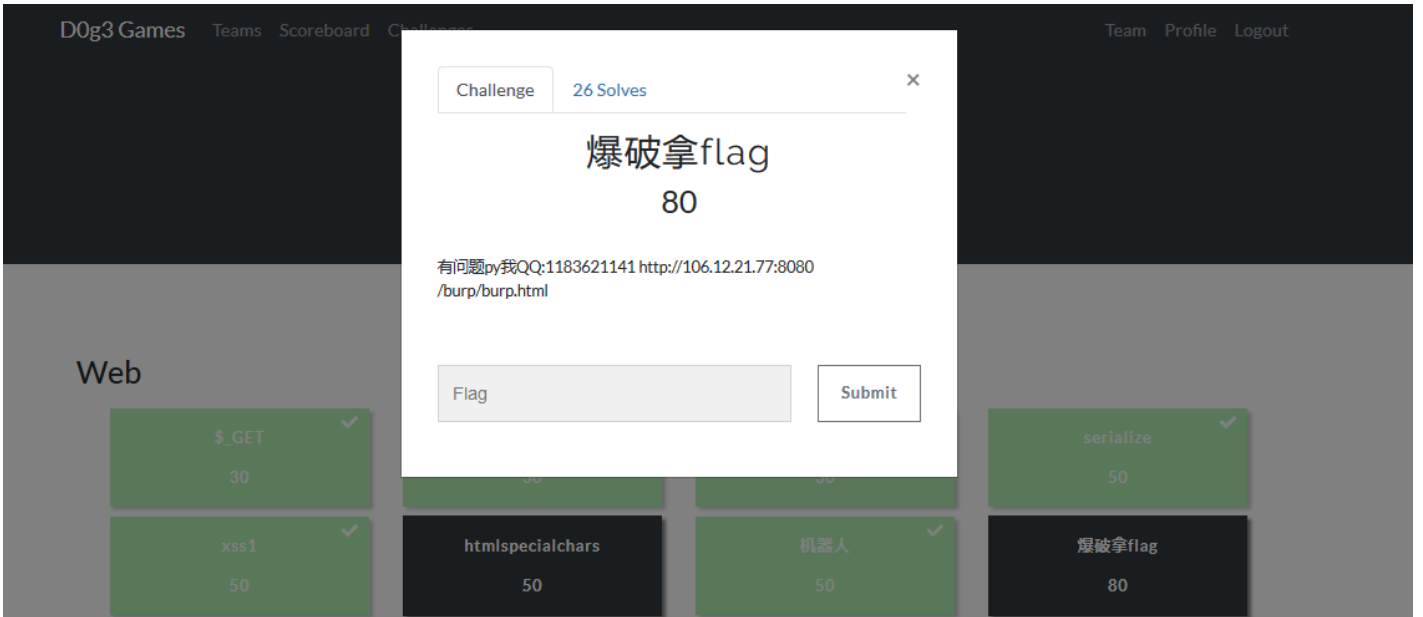


image.png

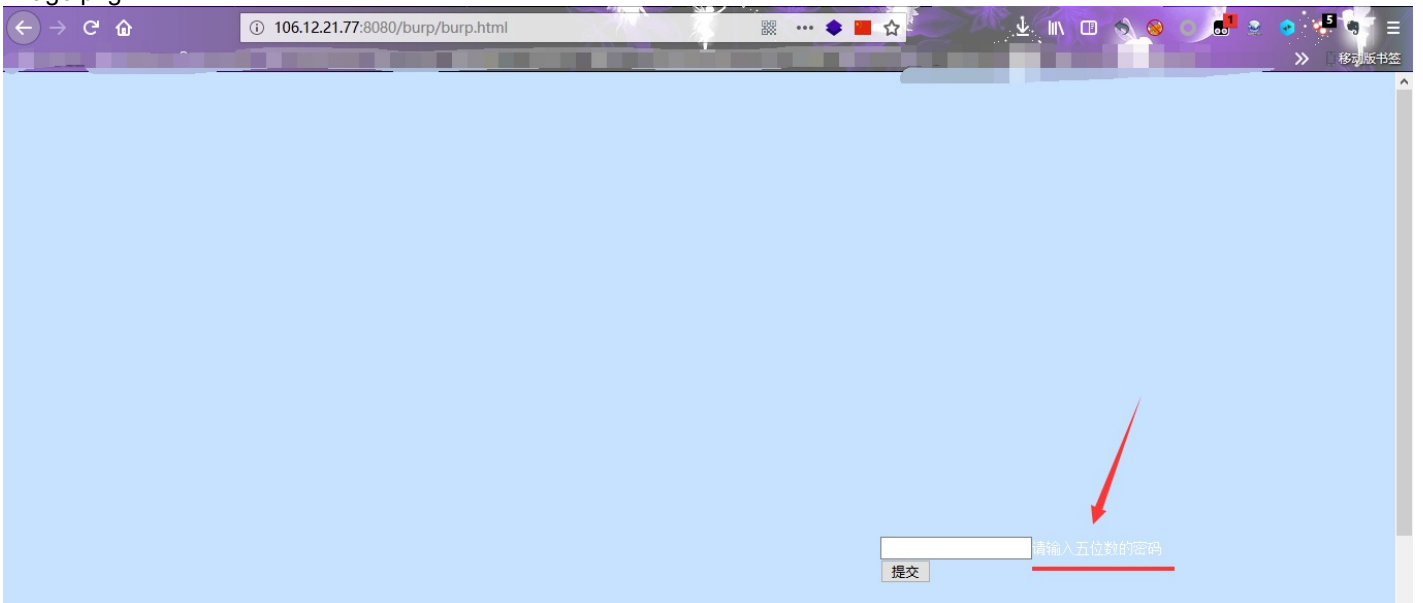


image.png

- 题目分析: 也是很简单的一道题, 5位数的密码, 废话不多说, 抓包爆破即可
- 解题方法:

Target Positions Payloads Options

### ? Payload Sets

You can define one or more payload sets. The number of payload sets depends on the attack each payload set, and each payload type can be customized in different ways.

Payload set:  Payload count: 100,000  
 Payload type:  Request count: 100,000

---

### ? Payload Options [Numbers]

This payload type generates numeric payloads within a given range and in a specified format.

Number range

Type:  Sequential  Random

From:   
 To:   
 Step:   
 How many:

Number format

image.png

耐心等待.....

Results Target Positions Payloads Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
12569	12568	200	<input type="checkbox"/>	<input type="checkbox"/>	216	
0		200	<input type="checkbox"/>	<input type="checkbox"/>	206	
1	0	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
2	1	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
3	2	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
4	3	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
5	4	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
6	5	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
7	6	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
9	8	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
10	9	200	<input type="checkbox"/>	<input type="checkbox"/>	206	
8	7	200	<input type="checkbox"/>	<input type="checkbox"/>	206	

Request Response

Raw Headers Hex

```

HTTP/1.1 200 OK
Server: nginx
Date: Wed, 31 Oct 2018 03:29:36 GMT
Content-Type: text/html
Connection: close
Vary: Accept-Encoding
X-Powered-By: PHP/5.4.41
Content-Length: 32
  
```

**D0g3{pass\_word\_is\_corret}**

? < + >

65805 of 100000

image.png

密码: 12568

## 9. 数字比较

- 题目地址：题目给的地址不能用，但是给了php文件，只能在自己环境里搭建咯。

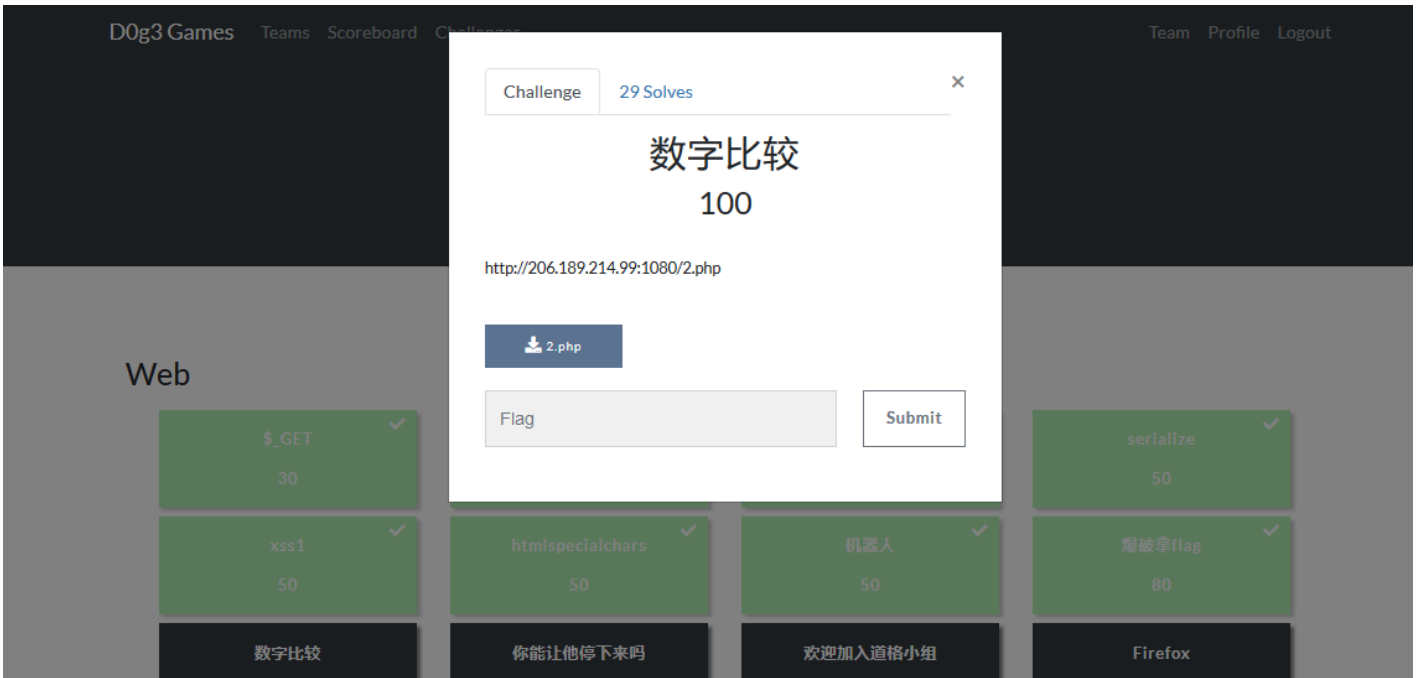


image.png

```
<?php
error_reporting(0);
function noother_says_correct($temp)
{
    $flag = "xxxxx";
    $one = ord('1'); //ord - 返回字符的 ASCII 码值
    $nine = ord('9'); //ord - 返回字符的 ASCII 码值
    $number = '3735929050';
    // Check all the input characters!
    for ($i = 0; $i < strlen($number); $i++)
    {
        // Disallow all the digits!
        $digit = ord($temp{$i});
        if ( ($digit >= $one) && ($digit <= $nine) )
        {
            // Aha, digit not allowed!
            return "flase";
        }
    }
    if($number == $temp)
    return $flag;
}
$temp = $_GET['password'];
echo noother_says_correct($temp);
?>
```

- 题目分析：这道题是数字比较，但是又不允许输入1到9数字，因此可以使用16进制来进行比较，即传入的参数值为3735929050的16六进制deadc0da，传入时候记得前面加0x。
- 解题方法：  
payload: ?password=0xdeadc0da

## 11. 欢迎加入道格小组(伪造数据包来源来源)

- 题目地址: <http://106.12.21.77:8080/referer/referer.php>

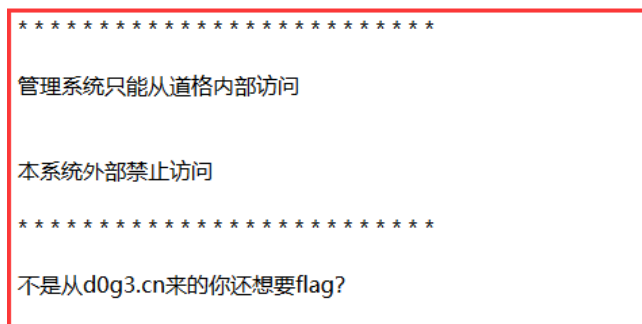
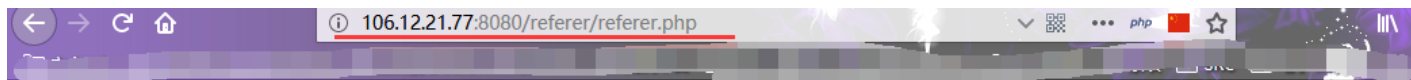


image.png  
抓取数据包

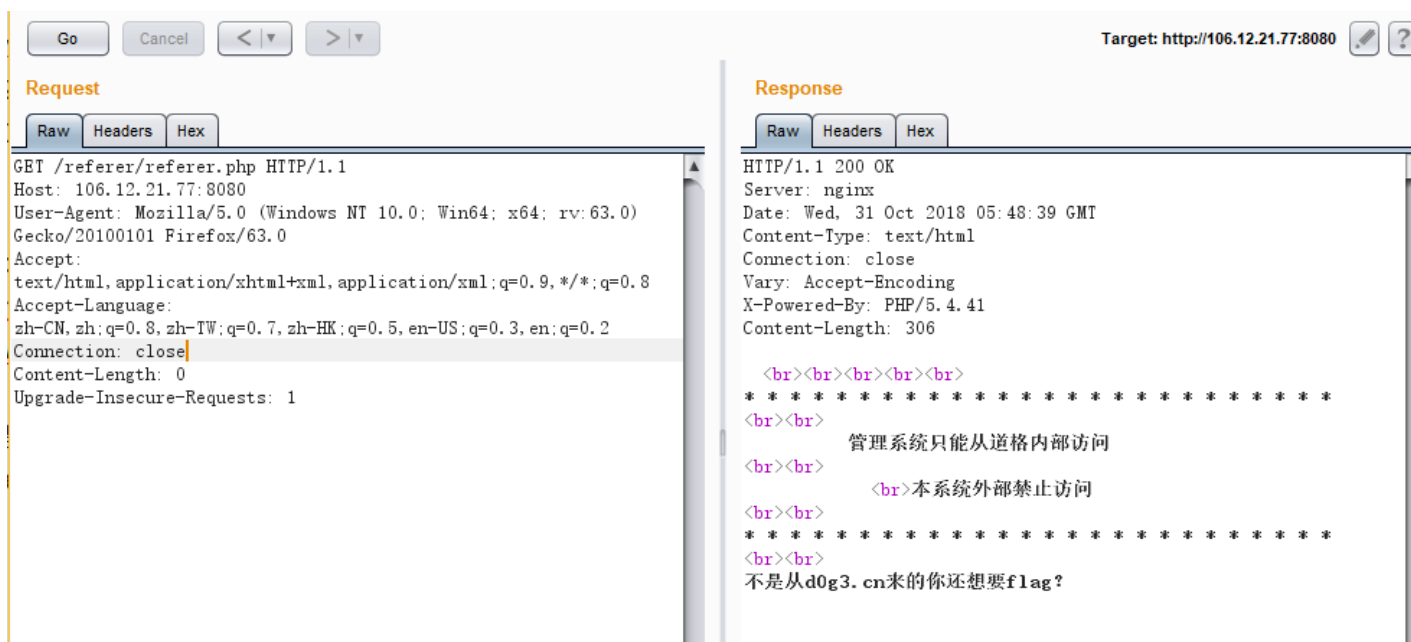


image.png

- 题目分析: 这么明显的提示, 就是伪造IP头部了, 又加之提示信息不是从d0g3.cn来的你还想要flag?, 很容易拿到flag。
- 解题方法: 伪造数据包头

Referer: d0g3.cn

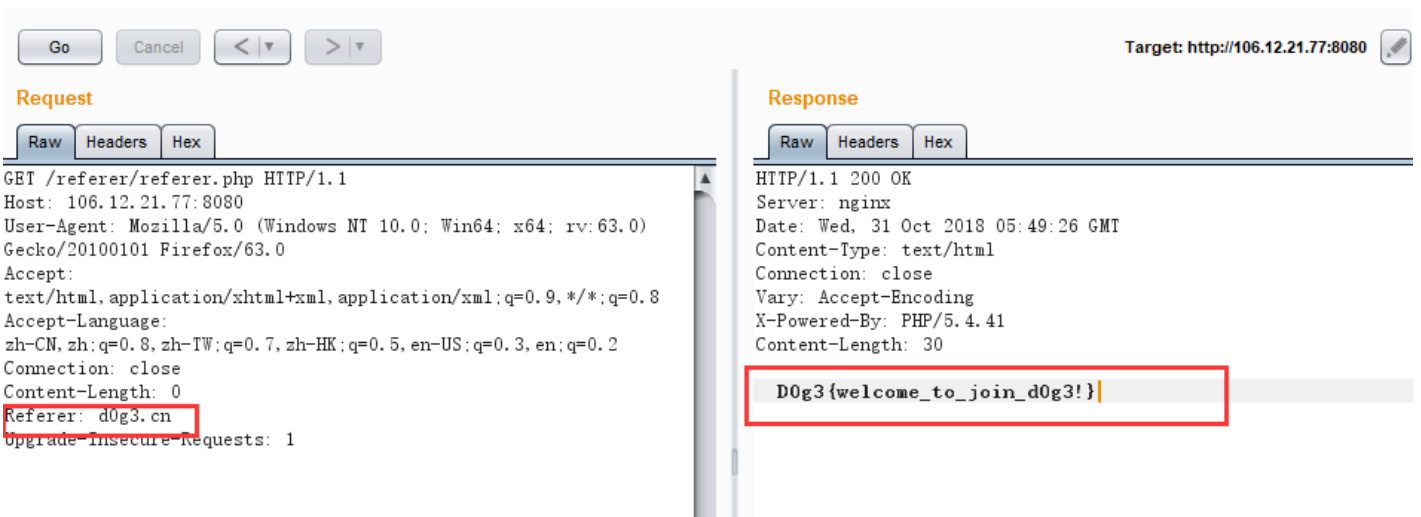


image.png

## 12. Firefox

- 题目地址: <http://106.12.21.77:8080/firefox.php>



image.png

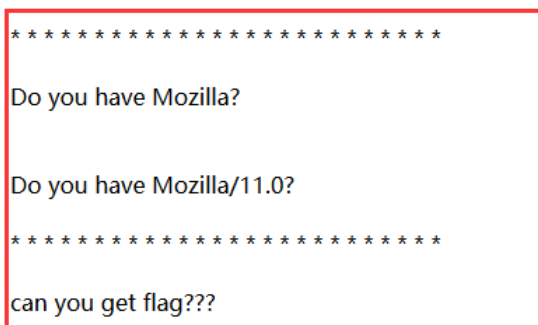
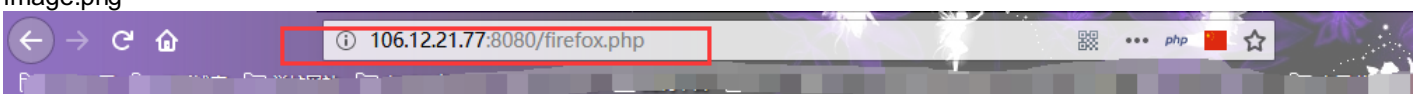
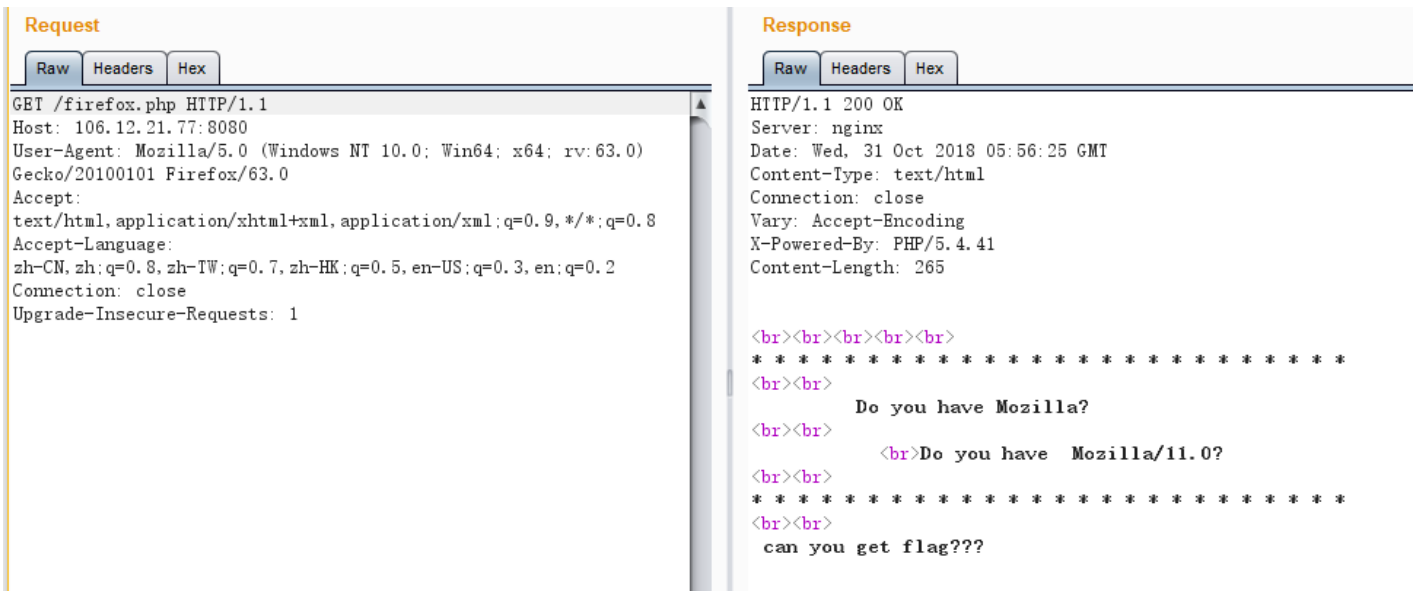


image.png



12

- 题目分析：看起来也很简单，就是伪造浏览器版本信息吧。
- 解题方法：修改浏览器版本信息为Mozilla/11.0，开始简单改个数字11，结果不行，后来仔细看提示，说只用构造为Mozilla/11.0，不用构造操作系统和协议。

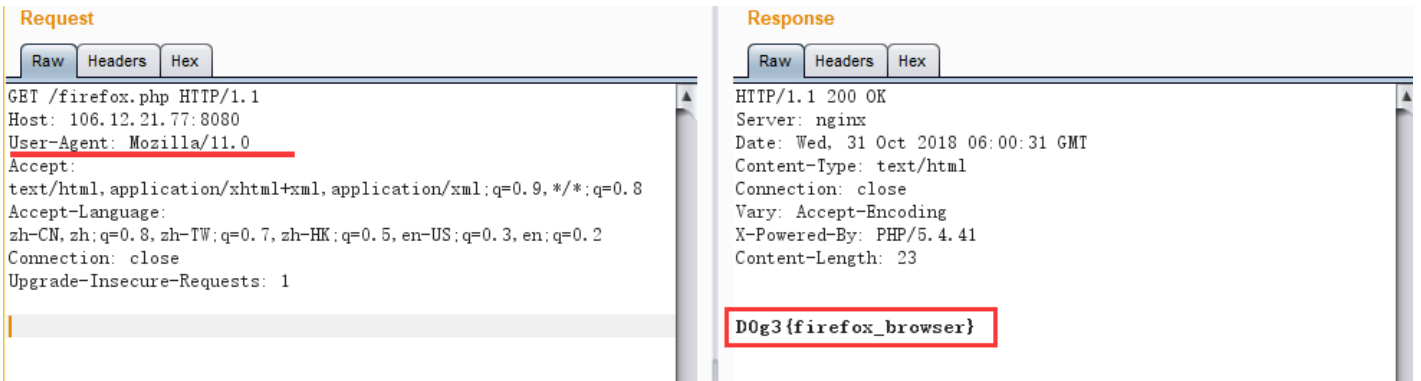


image.png

### 13. IP伪造

- 题目地址：<http://106.12.21.77:8080/x-forwarded-for.php>

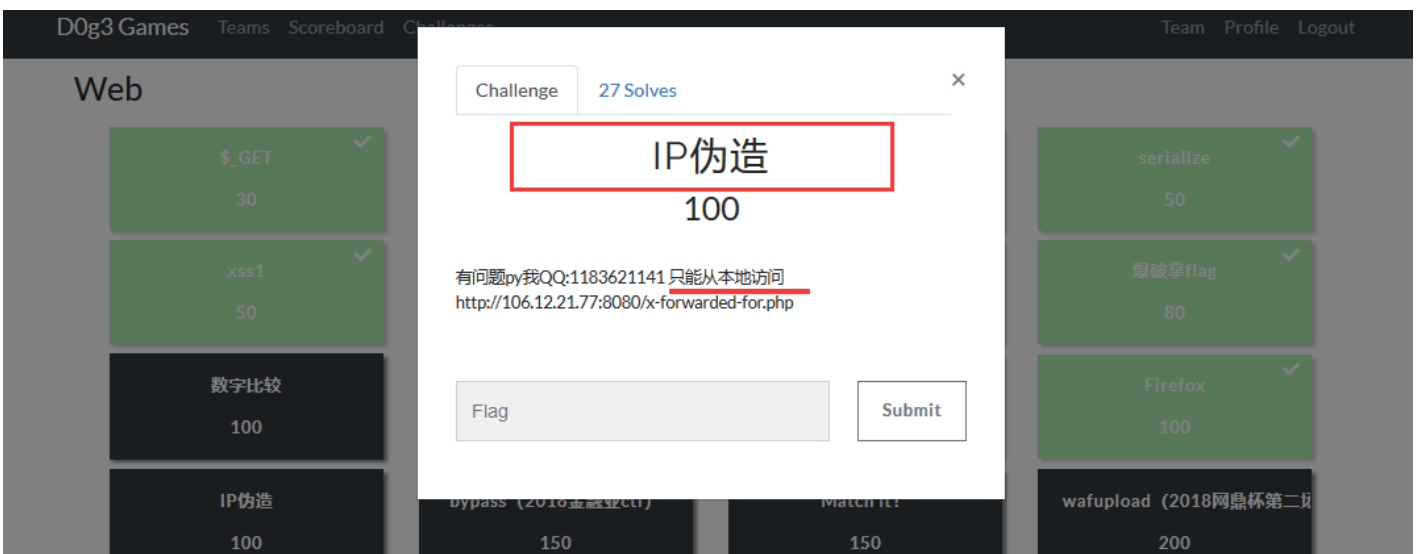


image.png

Request			Response		
Raw	Headers	Hex	Raw	Headers	Hex
<pre>GET /x-forwarded-for.php HTTP/1.1 Host: 106.12.21.77:8080 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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 Connection: close Upgrade-Insecure-Requests: 1</pre>			<pre>HTTP/1.1 200 OK Server: nginx Date: Wed, 31 Oct 2018 06:07:11 GMT Content-Type: text/html Connection: close Vary: Accept-Encoding X-Powered-By: PHP/5.4.41 Content-Length: 32  Your ip address is 222.66.55.241</pre>		

image.png

- 题目分析：很简单了，伪造IP

```
Client-IP: 127.0.0.1
X-Forwarded-For: 127.0.0.1
Host: 127.0.0.1
```

- 解题方法：抓取数据包修改数据包IP地址

Request			Response		
Raw	Headers	Hex	Raw	Headers	Hex
<pre>GET /x-forwarded-for.php HTTP/1.1 Host: 106.12.21.77:8080 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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 Connection: close X-Forwarded-For: 127.0.0.1 Upgrade-Insecure-Requests: 1</pre>			<pre>HTTP/1.1 200 OK Server: nginx Date: Wed, 31 Oct 2018 06:08:05 GMT Content-Type: text/html Connection: close Vary: Accept-Encoding X-Powered-By: PHP/5.4.41 Content-Length: 65  Your ip address is 127.0.0.1&lt;br&gt;&lt;br&gt;flag is : D0g3{x_forward_for}</pre>		

image.png

## 15. Match it! (匹配它)

- 题目地址：[http://206.189.214.99:1080/Pr3g\\_m4atch1/Pr3g\\_m4atch1.php](http://206.189.214.99:1080/Pr3g_m4atch1/Pr3g_m4atch1.php)





```
<?php
include 'flagi3hEre.php';
if ("POST" == $_SERVER['REQUEST_METHOD'])
{
    $password = $_POST['password'];
    if (0 >= preg_match('/^[[:graph:]]{12,}$/', $password))
    {
        echo 'Wrong Format';
        exit;
    }
    while (TRUE)
    {
        $reg = '/([[:punct:]]+|[[:digit:]]+|[[:upper:]]+|[[:lower:]]+)/';
        if (6 > preg_match_all($reg, $password, $arr))
            break;
        $c = 0;
        $ps = array('punct', 'digit', 'upper', 'lower');
        foreach ($ps as $pt)
        {
            if (preg_match("/[[:$pt:]]+/", $password))
                $c += 1;
        }
        if ($c < 3) break;
        if ("42" == $password) echo $flag;
        else echo 'Wrong password';
        exit;
    }
}
highlight_file(__FILE__);
?>
```

image.png

```
<?php
include 'flagi3hEre.php';
if ("POST" == $_SERVER['REQUEST_METHOD'])
{
    $password = $_POST['password'];
    if (0 >= preg_match('/^[[:graph:]]{12,}$/', $password))
    {
        echo 'Wrong Format';
        exit;
    }
    while (TRUE)
    {
        $reg = '/([[:punct:]]+|[[:digit:]]+|[[:upper:]]+|[[:lower:]]+)/';
        if (6 > preg_match_all($reg, $password, $arr))
            break;
        $c = 0;
        $ps = array('punct', 'digit', 'upper', 'lower');
        foreach ($ps as $pt)
        {
            if (preg_match("/[[:$pt:]]+/", $password))
                $c += 1;
        }
        if ($c < 3) break;
        if ("42" == $password) echo $flag;
        else echo 'Wrong password';
        exit;
    }
}
highlight_file(__FILE__);
?>
```

- 题目分析：一步一步代码审计吧。

```
0 >= preg_match('/^[[:graph:]]{12,}$/', $password)
//输入的内容必须为12个以上包括12，不包括空格和tab键

$reg = '/([[:punct:]]+|[[:digit:]]+|[[:upper:]]+|[[:lower:]]+)/';
//[[:punct:]] 任何标点符号， [[:digit:]] 任何数字， [[:upper:]] 任何大写字母， [[:lower:]] 任何小写字母

if (6 > preg_match_all($reg, $password, $arr))
//匹配到的次数要大于6次才能绕过。连续的大写或小写字母都只能算是一次。

$ps = array('punct', 'digit', 'upper', 'lower');
foreach ($ps as $pt)
{
    if (preg_match("/[[:$pt:]]+/", $password))
        $c += 1;
}
if ($c < 3) break;
//这个好理解，必须要有大小写字母，数字，字符内容三种与三种以上

if ("42" == $password) echo $flag;
//最后输入的值还要和42相等。
```

- 解题方法：根据以上匹配规则构造出满足需求的payload  
可以构造420.000000e-1 //0可以随意添加不影响数值，但是要超过12个字符

D0g3{Pr3g\_m4atch1\_c4nt\_st0p\_y0u!!!!}



image.png

太菜了，其他的有点思路但暂时没搞定。