

xctf_web区题解（入门区1--6）

原创

zmc曦 于 2021-12-13 10:42:17 发布 2482 收藏

分类专栏: [ctf web刷题记录](#) 文章标签: [web安全](#)

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_52185973/article/details/121900100

版权



[ctf web刷题记录](#) 专栏收录该内容

1 篇文章 0 订阅

订阅专栏

xctf_web区题解（新手区1—6）

ps:说明一下, 这次这个题解可能是比较是比较low的, 因为前一段时间在玩学校的vhdl来实现eda的大作业, 花了我一段比较长的时间, 另外之前一段时间再刷密码学的相关知识, 也没这么去了解这个web这个方向的, 对于小组内的blog我也就只能写写题解了, 以后可能做了一些项目会写一些其他有意思的东西

1.view_source

据题目描述, 这道题的flag应该是在网页的源代码中的, 但是鼠标右键点不了。那么这里有其他的解决方法, 我在这里主要说两个解决方法: 1.就是在网址上直接输入view-source: URL; 2.就是按F12进入开发者模式, 这里可以直接看见网页的源代码的(个人感觉这个很好用, 接下来的题我好像都用到了这个开发者模式)

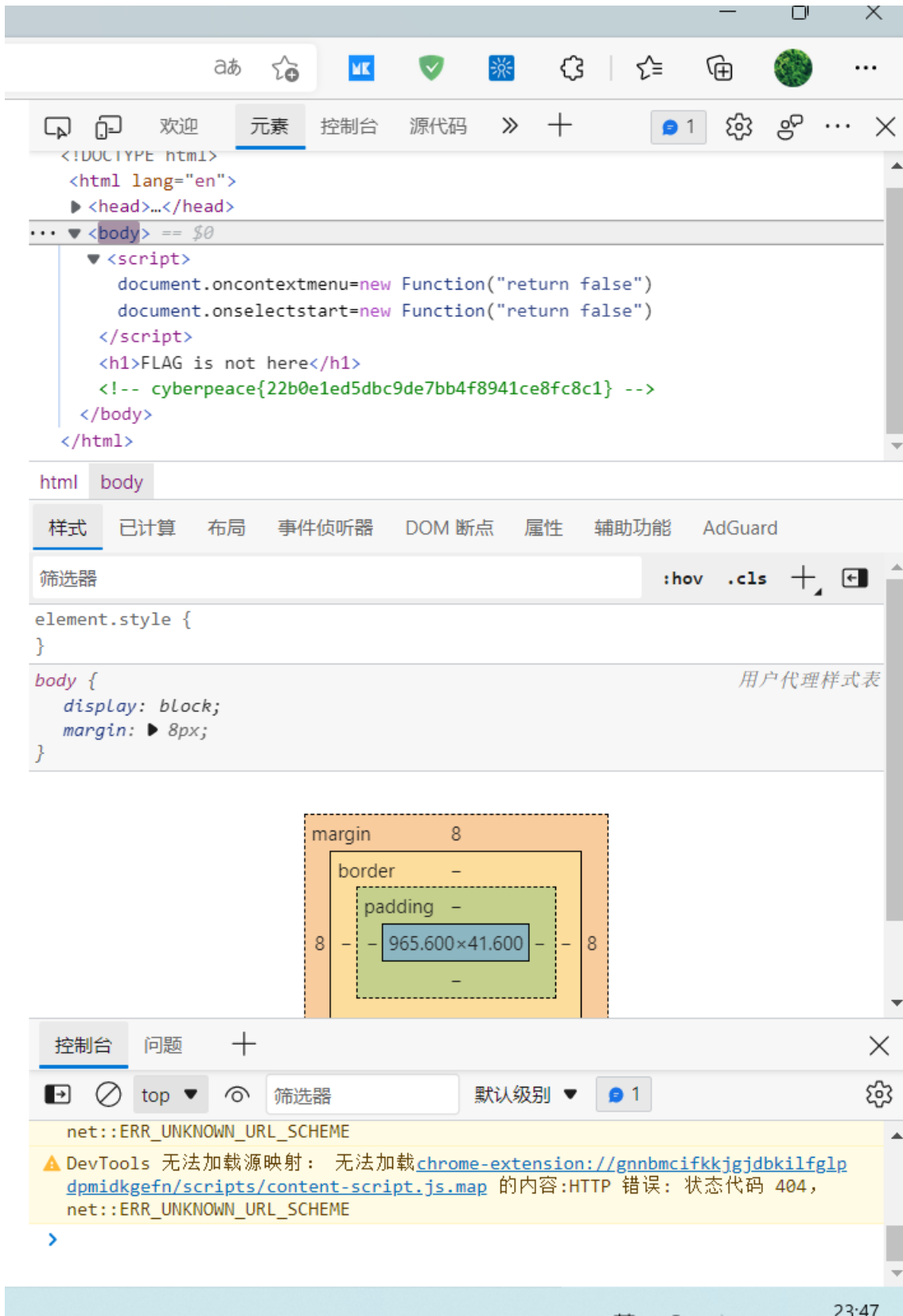
1.直接输入view-source:URL

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 [22b0e1ed5dbc9de7bb4f8941ce8fc8c1] -->
18
19 </body>
20 </html>
```

2. 直接在开发者模式查看

如下图：



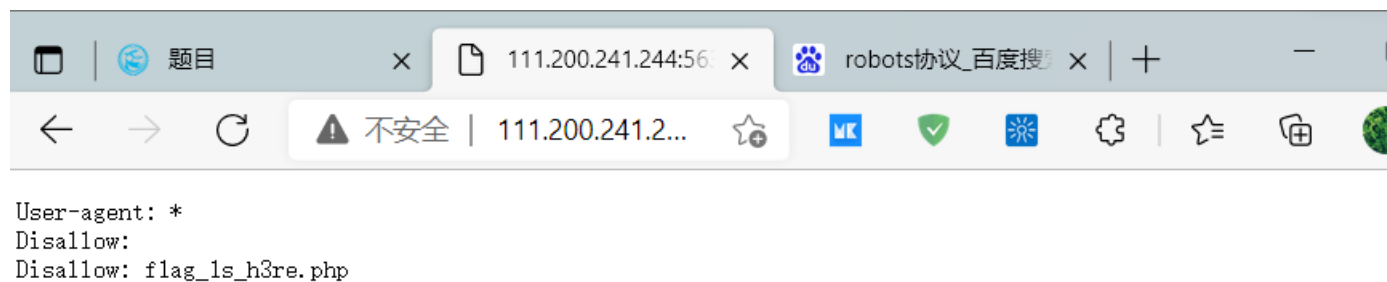
这个应该就是一个简单的签到题，就不过多细讲了。

2.robots

对于这个题目我简单的说一下**robots**，首先我简单的说一下什么是**Robots**协议，简单的来说**robots**协议就是一个存放在网站根目录下的ASCII编码的文本文件，所以也叫**robots.txt**，它的作用就是来告诉网络搜索引擎的漫游器（又叫网络蜘蛛），哪些地方是可以获取的，哪些地方是不能够获取的。

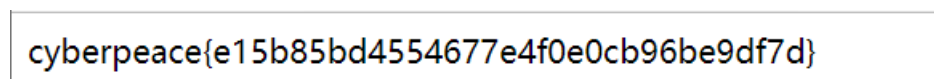
解题思路：本题很明显就是让你查看网站的robots.txt这个东西，所以查看就可以得到flag了，而对于查看网站的robots.txt，其实很简单就是在URL中输入robots.txt就可以查看网站的robots.txt了。

如下图：



然后你会发现***“flag_ls_h3re.php”*** 的字样在这个txt文档中，我们可以抱着试一试的态度将这个输入到URL中，然后就可以看见flag了

如图：



3. backup

据题目的描述就是要我们找出网页的备份，而且在题目给出的网站中你会发现是有关index.php的备份文件名，而对于网站的备份文件的后缀大多都是***“.git”、“.svn”、“.swp”、“.”、“.bak”、“.bash_history”、“.bkf”***放在URL的后面。

所以解题思路为：在URL后输入index.php.xx后缀名，然后就可以看到flag了

在URL中输入index.php.bak之后就可以看见下载了一个名为index.php.bak的文件，然后打开即可看见本题的flag

如下图



```
index.php.bak - 记事本
文件(F) 编辑(E) 格式(O) 视图(V) 帮助(H)
<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>
```

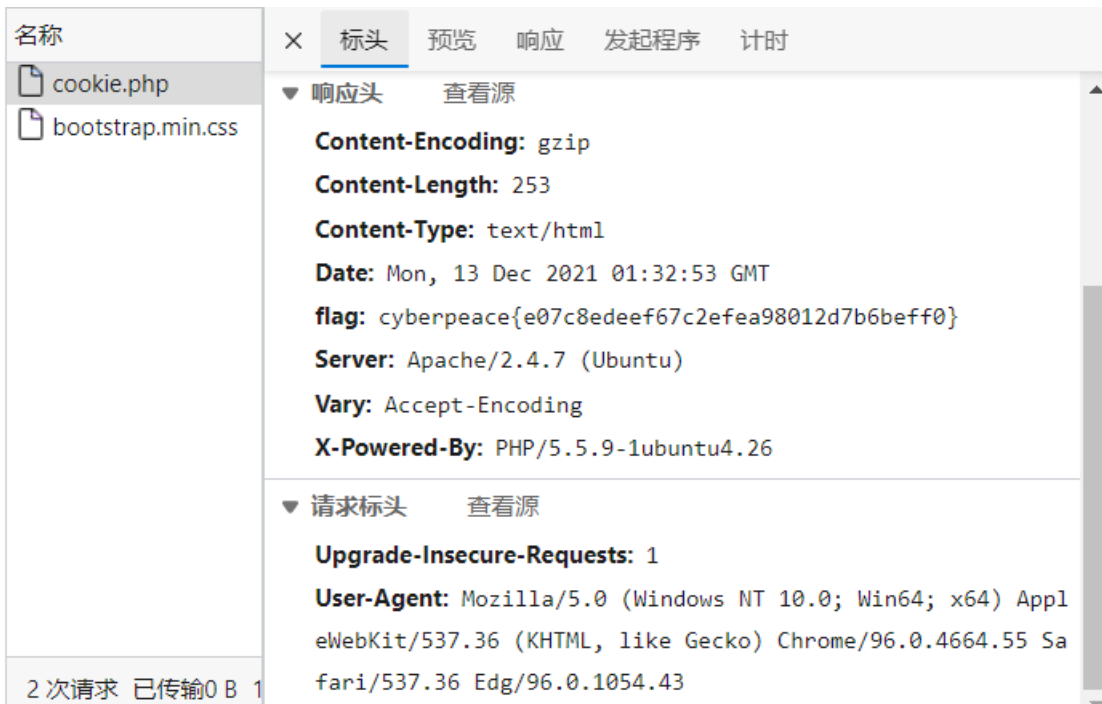
4.cookie

很明显就是一个flag放在cookie中的问题，看到cookie就可以得到这个flag。

首先简单的聊一下什么是cookie。**cookie**是某些网站为了辨别用户身份，进行**Session**跟踪而储存在用户本地终端上的数据（通常经过加密），由用户客户端计算机暂时或永久保存的信息，简单的来说就是储存在用户本地终端的数据，详细参考[cookie（储存在用户本地终端上的数据）_百度百科 \(baidu.com\)](#)

那么对于这道题目而言的话，就是查看**cookie**，但是如何查看？很简单就是打开**开发者模式**就可以了，然后选中“网络”这个选项就可以了，然后就开始监听网页的一些活动了，刷新一下网页即可，然后点击下面的**cookie**就可以看见响应的cookie是啥了，这个很明显就是那个**cookie.php**这个东西，然后将其输入到URL中去。此时再观察开发者的那个界面，就会发现出现了一个**cookie.php**的东西，然后在响应头就可以看见那个flag了

如图：

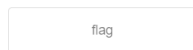


5.disabled_button

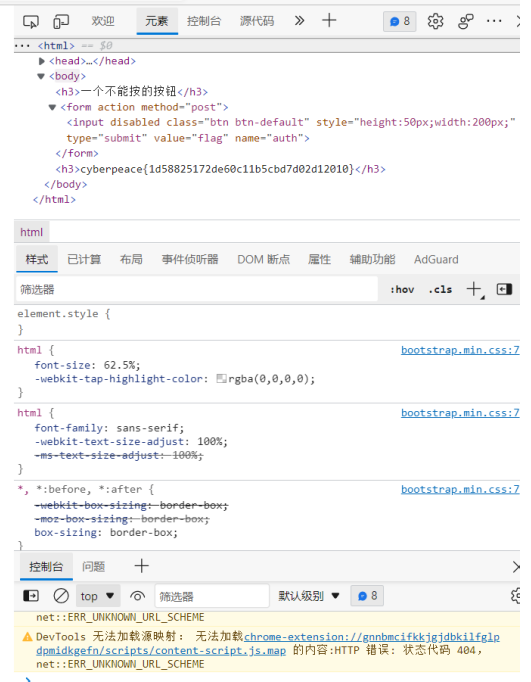
对于这题，应该就是改一下html的代码即可，将那个button的按键改为可以按下即可。那么如何改，同样还是打开开发者选项，查看网站的源代码，你会发现在有个disable的单词，就英文意思而言也是“不能的意思”，那么改一下捡起改为able，然后你就会发现，那个button就可以使用了，点击一下，就可以发现那个flag就会自己跳出来，提交即可。

结果如图：

一个不能按的按钮



cyberpeace{1d58825172de60c11b5cbd7d02d12010}



6.weak_auth

嗯，对于这题呢。其实非常的nice。从这个网页不难看出，这个就是需要将这个网页成功登陆后就可以得到相关的flag了。然后我就试了一两次，然后就登陆进去了（好吧，这是个1星题，密码也不会太难，但是直接输入“123456”就让我进去了是不是太草率了）。

下面我说一下，常规的破解方法，没错就是**爆破**（反正我是目前想不出来有啥好的方法）。对于这个的爆破有一个好的工具，那就是**burp suite**，我也是前几天才接触到这个东西的，用的还不是很习惯。首先好像要和浏览器搞一个代理，（推荐用Firefox似乎好搞一些，Microsoft edge好像不太好搞）至于如何建立我推荐一篇文章读者可以自己去阅读解决，本文就过多赘述了。

链接：[BurpSuite—代理和浏览器设置](#)

在配置完之后，打开burp suite，然后在proxy将intercept打开，然后再题目的容器中输入相关的信息，然后你就会发现在burp suite中会收到数据包。在点击intruder，进入爆破界面。首先输入攻击的ip，以及相应的port；然后再positions中将一开始发过来的数据包copy过来，通过**add**，**clear**操作选择爆破的对象。这里我们选择**password**，然后**payload**中添加字典，然后开始可以爆破了。爆破之后，然后点击123456的那条密码，然后就可以看见response中有一个flag。ok，破解成功。（因为事先就知道了密码，所以就是验证一下，至于burp suite这个软件，个人还是不怎么怎么会用）

附爆破图：

The screenshot shows the Burp Suite interface during an intruder attack. The main window displays the results of the attack, with a table listing the requests and their corresponding responses. The first request (index 1) is highlighted, showing a successful login with a status of 200. The response body is displayed in the 'Response' tab, showing an HTML page with a title 'weak auth' and a body containing a flag: 'cyberpeace{f45181e04bd37c16d843b75507a9509c}<!--maybe you need a dictionary-->'. A red circle highlights the flag in the response body. The interface also shows the 'Request' and 'Response' tabs, and a search bar at the bottom.

Request	Payload	Status	Error	Timeout	Length	Comment
0						
1	123456	200			437	

```
11 <html lang="en">
12 <head>
13 <meta charset="UTF-8">
14 <title>
15   weak auth
16 </title>
17 </head>
18 <body>
19   cyberpeace{f45181e04bd37c16d843b75507a9509c}<!--maybe you need a dictionary-->
20 </body>
21 </html>
```