Xp0int2016新生杯CTF-writeup

原创

sherlly666 于 2016-10-30 18:52:51 发布 3212 收藏 1

分类专栏： CTF 文章标签： CTF writeup

本文链接：https://blog.csdn.net/github_35681219/article/details/52973422

先说说自己参加这次比赛的感受吧，虽然说只是一次校内的比赛，但是它于我的意义却远不止如此。CTF这个字眼，之前对于我来说一直是“可远观而不可亵玩焉”的感觉，这次算是初次接触，为期两周的比赛，从前端网页的基础知识到底层汇编、逆向、破解等，比赛所涉及的知识网络无疑是庞大的，可以说是自学效率最高的一次= =，话不多说，写了下wp,也算是记录一下这次经历吧。

比赛地址：http://www.giantbranch.cn:8889/

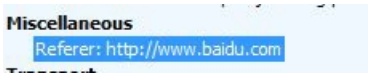
Web

0x00 where r u from

地址：http://pyzpyjy.cn/refer/

题目考察http头文件知识，按照给的提示将Referer地址修改为www.baidu.com即可。

这里用了Fiddler工具截获网页数据，按如下修改：



flag:{**Xp0int-referer**}

0x01 xss

地址：http://pyzpyjy.cn/xss/

题目考察最基本的xss攻击，在填写框中填入`<script>alert('test')</script>`，提交出flag.

flag:{**Xp0int-xssinteresting**}

0x02 Get flag

地址：http://123.207.1.235/ctf/getflag.php

这里考察了http地址中参数的直接传递，将http://123.207.1.235/ctf/getflag.php?flag=0修改为http://123.207.1.235/ctf/getflag.php?flag=1即可。

flag{**Xp0int_is_here**}

0x03 无脑前端

地址：http://www.giantbranch.cn:8889/static/uploads/d2b01d2bc2dbe29712d11839b927b19b/lwh-.html

直接查看页面源代码，发现这么一行：

```
    alert("ZmxhZyhYcDBpbnRBbm5vbW9yaXR5Sk5VKQo= (p.s.可以试一试base64解码)");
```

把`ZmxhZyhYcDBpbnRBbm5vbW9yaXR5Sk5VKQo=`放进在线解密网站，直接出结果。
flag(**Xp0intAnnomorityJNU**)

0x04 PHP~
地址：http://www.lynnlinlin.cn/Xp0int/index.php
按提示查看index.php.txt，发现如果id参数进行urldecode解码后为Xp0int233就会显示flag，尝试将Xp0int进行url编码，发现在线编码无法成功，转用手动编码，查表得`%58%70%30%69%6E%74%32%33%33`，将转码后的Xp0int233输入地址：http://www.lynnlinlin.cn/Xp0int/index.php?id=%58%70%30%69%6E%74%32%33%33，加载后提示不通过，发现http在发送数据时会自动进行一次url转码，所以应该进行两次转码，尝试将`%58%70%30%69%6E%74%32%33%33`转码为`%2558%2570%2530%2569%256E%2574%2532%2533%2533`并输入http://www.lynnlinlin.cn/Xp0int/index.php?id=%2558%2570%2530%2569%256E%2574%2532%2533%2533，出flag。
flag: {**Xp0int666666**}

0x05 cookie
地址：http://123.207.1.235/ctf/setcookie.php
根据题目提示修改cookie，这里用了火狐的Tamper Data插件进行抓包，修改cookie的flag值为1，提交出flag。
flag{**Xp0int_is_a_great_team**}

0x06 JsFuck
地址：http://123.207.1.235/ctf/jsfuck.html
题目给的是js的一种编码，代码全部由`[]()!`组成，将代码复制进控制台，提交出结果。
flag{**Xp0int_welcomes_you**}

0x07 Login Brute Force
地址：http://pyzpyjy.cn/session/
呃…这一题就有点坑了，题目给了用户名和密码的登录界面，由于题目的提示，一开始跑偏往爆破的方向，结果死活跑不出结果，后来看了网上一篇writeup，有感而发试了一下admin目录，居然出了提示请先登录的界面，接着尝试注入，出flag，看来题目也不能尽信呐…
flag{**Xp0int-wobeikengle**}

Crypto

0x01 Crypto1
地址：http://pyzpyjy.cn/zhujuan/
典型的猪圈密码…百度对照表得zhu，填入框中得flag。
flag:{**Xp0int-zhujuanmima**}

0x02 Morse
就是简单的摩斯密码，随便放进在线解密网站，出结果。（附带吐槽一句，明明答案是对的，结果交上去一直不通过，真的是…）
FLAG **XPINTMORSECODE**

0x03 你猜
题目：ESZDCTF 852 IUHNM GHYTFVB
这道挺有意思的，按键盘布局加密。

flag:**NICE**

0x04 Go
题目：BABAB ABBBA ABBAB ABAAA ABBAA BAABA AABBA ABBAB
由题目提示可知是培根密码，查表得flag。
flag:**xpointgo**

0x05 搞那么多干嘛
题目：
JZVEKMS2IRJGYTL2IEYVSVC2NRHHUUJSLF5FE3CNPJETAWKULJVE4R2VGBHEIUTMJZVGWMKPKRKTGTSENMZU6VCWNBHG2RJQLJCE2MCOI5KTGWKUKV4E26SJGFMVIVJQJZWUS6SNKRJG2TSUKEYU4VCNPBHEOUJQJZCFK6KONJTTAWSUKEZU4R2RGNHUIVTIJZWVCMKOKRSGQTSHKUYE6RCNO5GTEUJ5
这道题真的是…脑洞无限。
先base32转码，
4E6A45325A44526C4D7A413159545A6C4E7A5132597A526C4D7A493059545A6A4E4755304E44526C4E6A6B314F5455334E446B334F5456684E6D45305A444D304E475533595455784D7A4931595455304E6D497A4D54526D4E5451314E544D784E4751304E4455794E6A67305A5451334E4751334F4456684E6D51314E5464684E4755304F444D774D32513D
再ASCII转码，
NjE2ZDRlMzA1YTZlNzQ2YzRlMzl0YTZjNGU0NDRlNjk1OTU3NDk3OTVhNmE0ZDM0NGU3YTUxMzl1YTU0NmIzMTRmNTQ1NTMxNGQ0NDUyNjg0ZTQ3NGQ3ODVhNmQ1NTdhNGU0ODMwM2Q=
看到=标志进行base64转码，
616d4e305a6e746c4e324a6c4e444e69595749795a6a4d344e7a51325a546b314f5455314d4452684e474d785a6d557a4e48303d
再ASCII转码，
amN0ZntlN2JlNDNiYWIyZjM4NzQ2ZTk1OTU1MDRhNGMxZmUzNH0=
再base64转码，出flag
jctf{**e7be43bab2f38746e9595504a4c1fe34**}

Re

0x01 请输入flag
题目给了个.zip格式的文件，拖进OD分析，跑一下直接栈窗口出结果= =
flag:**we1c0_2_xp0int**

0x02 first Android
先将apk解压，找出其中的classes.dex，用dex2jar转为jar，然后这里就卡了，把jar放在jd-gui中分析半天没出结果= =，结果发现想太多，直接把apk拖进UltraEdit，查找字符串flag，出结果。（一口老血…）
jctf{**y0u_le6rn_that**}

0x03 Read ASM

题目：

```
int main(int argc, char const *argv[])
{
char input[] = {0x0, 0x67, 0x6e, 0x62, 0x63, 0x7e, 0x74, 0x62, 0x69, 0x6d,
0x55, 0x6a, 0x7f, 0x60, 0x51, 0x66, 0x63, 0x4e, 0x66, 0x7b,
0x71, 0x4a, 0x74, 0x76, 0x6b, 0x70, 0x79, 0x66 , 0x1c};
func(input, 28);
printf("%s\n",input+1);
return 0;
}
```

func函数的asm代码如下:

00000000004004e6 :

4004e6: 55 push rbp

4004e7: 48 89 e5 mov rbp,rsp

4004ea: 48 89 7d e8 mov QWORD PTR [rbp-0x18],rdi

4004ee: 89 75 e4 mov DWORD PTR [rbp-0x1c],esi

4004f1: c7 45 fc 01 00 00 00 mov DWORD PTR [rbp-0x4],0x1

4004f8: eb 28 jmp 400522

4004fa: 8b 45 fc mov eax,DWORD PTR [rbp-0x4]

4004fd: 48 63 d0 movsxd rdx,eax

400500: 48 8b 45 e8 mov rax,QWORD PTR [rbp-0x18]

400504: 48 01 d0 add rax,rdx

400507: 8b 55 fc mov edx,DWORD PTR [rbp-0x4]

40050a: 48 63 ca movsxd rcx,edx

40050d: 48 8b 55 e8 mov rdx,QWORD PTR [rbp-0x18]

400511: 48 01 ca add rdx,rcx

400514: 0f b6 0a movzx ecx,BYTE PTR [rdx]

400517: 8b 55 fc mov edx,DWORD PTR [rbp-0x4]

40051a: 31 ca xor edx,ecx

40051c: 88 10 mov BYTE PTR [rax],dl

40051e: 83 45 fc 01 add DWORD PTR [rbp-0x4],0x1

400522: 8b 45 fc mov eax,DWORD PTR [rbp-0x4]

400525: 3b 45 e4 cmp eax,DWORD PTR [rbp-0x1c]

400528: 7e d0 jle 4004fa

40052a: 90 nop

40052b: 5d pop rbp

40052c: c3 ret

主要就是要读懂汇编码= =，具体见另一篇博客一道逆向CTF题-readasm详解，放一下func的源程序，

```
void func(char input[],int num)
{
    int i = 1;
    while(i <= num)
    {
        *(input[0]+i) ^= i;
        i++;
    }
}
```

跑一下出结果：
flag{**read_asm_is_the_basic**}

0x04 抛骰子
题目：http://pan.baidu.com/s/1slTHxcp
下载下来放进OD里，运行发现只有每次准确扔出特定的点数，程序才会进行下一步操作，否则退出。在每一次程序作出判断前下断，将跳转到失败代码处的汇编语句改掉，如jz改为jnz，重复几次后出结果。
flag{**9a9689dbd47a1fd3fc0bf17d60edf545**}

Misc

0x00 签到题
这没啥好说的，会用微信就可以了…
flag{**Xp0int_123456789**}

0x01 Xp0int-Stegano
典型的图像题…把.png拖进UltraEdit，拖到最后看见flag。
flag:**X-p0int122333**

0x02 Xp0int-Change
这道题真的是困惑了很久…由于字符串均由\u开头，考虑unicode编码，解码后出来这个…`-2:dcug86\ozj|r9YJCycY72NYToQFe7QJL;`试了很多种解密都失败了，最后看了网上古典密码的资料还有几篇加密相关的writeup，想到应该是凯撒密码的变形，写个脚本进行128次轮转爆破…

```
str="-2:dcug86\\ozj\\|r9YJCycY72NYToQFe7QJL;"

for p in range(127):
    str1 = '\n'
    for i in str:
        temp = chr((ord(i)+p)%127)
        if 32<ord(temp)<127 :
            str1 = str1 + temp
            feel = 1
        else:
            feel = 0
            break
    if feel == 1:
        print(str1)
```

出来结果是：

```
-2:dcug86\ozj\|r9YJCycY72NYToQFe7QJL;

.3;edvh97]p{k]}s:ZKDzdZ83OZUpRGf8RKM<

/4<fewi:8^q|l^~t;[LE{e[94P[VqSHg9SLN=

!&.XWi[,*Pcn^Ppf-M>7mWM+&BMHcE:Y+E>@/

"'/YXj\-+Qdo_Qqg.N?8nXN,'CNIdF;Z,F?A0

#(0ZYk].,Rep`Rrh/O@9oYO-(DOJeG<[-G@B1

$)1[Zl^/-SfqaSsi0PA:pZP.)EPKfH=\.HAC2

%*2\[m_0.TgrbTtj1QB;q[Q/*FQLgI>]/IBD3

&+3]\n`1/UhscUuk2RC<r\R0+GRMhJ?^0JCE4

',4^]oa20VitdVvl3SD=s]S1,HSNiK@_1KDF5

(-5_^pb31WjueWwm4TE>t^T2-ITOjLA`2LEG6

).6`_qc42XkvfXxn5UF?u_U3.JUPkMBa3MFH7

*/7a`rd53YlwgYyo6VG@v`V4/KVQlNCb4NGI8

+08base64ZmxhZzp7WHAwaW50LWRmODc5OHJ9

,19cbtf75[nyi[{q8XIBxbX61MXSnPEd6PIK:
```

看到最后的base64字眼，把ZmxhZzp7WHAwaW50LWRmODc5OHJ9进行base64解密，得flag。
flag:{**Xp0int-df8798r**}

0x04 隐写术
题目给了个doc文件，打开后只显示flag在哪呢？把文件重新下载为zip格式，打开发现有几个目录，打开flag->word->document.xml，找到flag。
Flag{**Xp0int_y0u_ar4_5o_clever**}

这次比赛账号是brzhj25，成绩如下，还是要多多努力呀！

Place	Team	Score
1	首长说太容易了于是我过来了	2290
2	cheam	2290
3	我也是江职的	2110
4	brzhj25	1740
5	tiffany	1460