170616 逆向-flag_check

原创

奈沙夜影 于 2017-06-17 02:20:36 发布 803 收藏
分类专栏： CTF

1625-5 王子昂 总结《2017年6月16日》【连续第257天总结】

A.实验吧-check flag

B.题目是一个网页：http://ctf5.shiyanbar.com/qwctf/flag-checker.html

打开直接弹窗问Flag，F12查看源代码发现输入与一个很长很恐怖的式子进行校验：

```
if (a.length==47&&a[11]-a[5]%a[1]*a[12]%a[14]-a[6]*a[7]-a[24]+a[10]%a[13]+a[2]*a[23]+a[21]+a[15]%a[3]%a[19]-a[20]*a[0]%a[17]+a[18]-a[22]+a[8]+a[4
a[26]-a[1]%a[11]+a[2]+a[25]+a[13]==-42&&a[34]+a[27]+a[29]-a[10]*a[13]*a[20]%a[31]+a[2]*a[23]-a[0]*a[8]-a[6]*a[9]-a[19]%a[3]+a[15]%a[1]-a[25]-a[26
a[12]+a[11]%a[9]%a[3]%a[0]*a[8]%a[2]+a[5]%a[17]+a[15]-a[10]%a[18]-a[6]-a[1]-a[19]*a[7]%a[14]==-76&&a[6]%a[29]+a[4]+a[18]%a[22]+a[16]+a[30]-a[19]*
a[3]%a[7]%a[8]*a[9]-a[0]+a[35]==-129&&a[1]*a[3]*a[5]%a[6]-a[2]%a[0]+a[4]-a[7]==18&&a[19]+a[20]+a[4]+a[0]-a[17]-a[8]-a[2]*a[7]+a[18]-a[14]-a[3]-a[
a[0]%a[7]%a[9]%a[6]-a[2]==-157&&a[9]%a[5]-a[11]+a[7]-a[0]%a[10]*a[4]%a[3]+a[1]-a[6]%a[8]+a[2]==99&&a[1]+a[4]%a[0]%a[3]*a[7]%a[6]-a[8]-a[2]+a[5]==
a[21]+a[28]*a[18]-a[37]+a[38]+a[11]%a[20]+a[9]-a[32]-a[0]*a[14]+a[33]*a[12]+a[24]-a[19]+a[6]==4&&a[2]-a[1]-a[3]+a[0]==-12&&a[6]%a[25]%a[17]+a[24]
a[20]*a[4]+a[30]+a[18]*a[16]-a[9]%a[1]%a[19]==132&&a[17]-a[11]+a[1]-a[0]%a[5]*a[12]%a[13]+a[4]%a[14]-a[10]-a[15]%a[8]%a[7]+a[6]-a[2]%a[16]+a[9]+a
a[6]*a[1]-a[14]+a[18]+a[10]%a[8]-a[17]==-43&&a[11]*a[2]%a[19]*a[6]-a[14]+a[32]*a[1]%a[28]-a[3]%a[27]-a[4]+a[13]+a[24]%a[12]-a[10]+a[23]-a[15]%a[0
a[18]%a[13]+a[10]+a[0]%a[5]-a[23]+a[15]*a[21]%a[20]+a[9]+a[7]-a[19]*a[2]-a[24]+a[1]*a[14]+a[6]%a[4]%a[8]%a[3]-a[22]-a[12]==-130&&a[1]-a[3]-a[6]*a
a[21]+a[34]%a[15]+a[9]+a[19]%a[13]+a[36]+a[18]%a[11]*a[12]%a[30]+a[29]+a[31]%a[17]-a[5]*a[24]*a[20]%a[8]%a[22]-a[4]-a[25]%a[10]-a[6]-a[3]-a[0]*a[
a[20]%a[27]*a[13]+a[30]%a[14]*a[1]%a[6]+a[37]%a[4]-a[26]*a[21]-a[7]-a[36]-a[16]*a[0]+a[28]*a[34]*a[42]*a[41]%a[9]-a[15]+a[19]*a[24]+a[11]*a[17]+a
a[21]*a[36]*a[4]%a[11]+a[32]-a[29]%a[34]-a[2]+a[20]%a[1]+a[10]-a[33]+a[0]%a[19]-a[22]+a[8]+a[13]%a[31]+a[17]%a[24]%a[7]+a[26]-a[3]%a[14]%a[12]*a[
a[38]+a[23]+a[6]%a[28]%a[27]%a[14]%a[39]+a[13]+a[32]+a[40]-a[4]-a[8]%a[11]%a[25]*a[31]+a[20]*a[24]%a[29]%a[34]*a[30]%a[33]%a[5]-a[26]*a[18]%a[2]+
a[24]*a[16]+a[10]%a[2]%a[7]*a[6]+a[23]+a[0]+a[4]*a[22]-a[30]+a[12]-a[17]%a[5]%a[1]*a[15]-a[19]*a[20]-a[3]-a[9]-a[13]+a[25]*a[18]%a[14]+a[26]%a[28
a[3]%a[10]+a[23]%a[12]+a[37]*a[29]%a[2]+a[30]-a[22]+a[32]%a[34]+a[33]+a[8]%a[26]*a[11]*a[15]%a[40]%a[5]%a[19]-a[21]+a[43]%a[6]*a[35]+a[27]==-76&&
a[5]+a[0]==50&&a[44]-a[24]+a[25]%a[30]%a[41]*a[3]-a[23]+a[20]*a[38]+a[15]-a[43]+a[8]-a[29]*a[9]+a[27]+a[33]-a[39]*a[18]%a[0]+a[7]-a[6]-a[42]-a[2]
a[1]==-44&&a[9]%a[3]%a[7]%a[0]%a[4]-a[2]-a[11]-a[12]+a[6]-a[5]%a[10]+a[8]-a[1]==-187&&a[8]%a[7]+a[6]-a[14]-a[4]%a[17]+a[11]-a[12]%a[5]%a[2]+a[15]
a[5]+a[9]+a[6]%a[7]*a[8]%a[14]==-22&&a[4]+a[16]+a[10]+a[5]-a[7]-a[11]-a[9]%a[13]-a[1]-a[12]%a[2]*a[14]%a[8]%a[6]+a[3]-a[15]%a[0]==97&&a[0]*a[1]%a
a[1]==91&&a[5]*a[4]%a[12]+a[18]+a[27]+a[22]+a[21]-a[10]-a[25]-a[20]%a[7]+a[14]%a[17]%a[23]+a[19]*a[13]%a[26]-a[1]*a[3]%a[8]+a[24]-a[6]+a[16]+a[0]
a[2]+a[28]%a[17]*a[1]%a[0]-a[3]%a[16]-a[12]*a[20]+a[29]+a[27]%a[24]%a[19]-a[13]==-46&&a[21]-a[36]-a[16]+a[28]-a[3]%a[1]+a[35]-a[8]-a[30]+a[29]%a[
a[6]*a[37]%a[17]*a[25]%a[9]%a[22]-a[0]*a[13]%a[32]==184&&a[38]*a[34]%a[30]-a[31]+a[26]-a[27]-a[16]%a[0]-a[18]%a[24]%a[29]%a[12]+a[20]-a[15]*a[7]+
a[32]%a[14]%a[11]*a[21]%a[5]+a[33]==167&&a[13]-a[6]%a[5]-a[0]%a[9]+a[21]+a[23]+a[18]%a[17]%a[16]%a[7]-a[20]%a[1]*a[15]%a[19]%a[8]*a[2]%a[22]-a[14
a[36]+a[35]+a[29]+a[34]-a[31]-a[5]+a[41]%a[3]*a[13]*a[10]%a[12]-a[21]*a[38]%a[24]-a[46]+a[33]%a[4]%a[11]*a[40]%a[44]+a[28]-a[22]%a[30]+a[8]-a[19]
a[42]+a[26]%a[39]%a[10]+a[28]%a[27]-a[7]-a[30]-a[19]%a[2]+a[32]%a[0]-a[6]+a[11]-a[13]%a[35]*a[29]-a[4]*a[24]*a[37]%a[40]+a[31]%a[33]*a[38]-a[21]+
a[15]+a[23]*a[17]*a[27]%a[8]-a[14]%a[22]-a[29]*a[5]%a[25]+a[4]%a[10]-a[19]%a[7]%a[12]%a[11]+a[20]+a[31]+a[18]+a[21]*a[30]%a[1]%a[28]+a[0]%a[26]+a
    alert("correct!");
```

分析观察，以&&作为分隔符，然后按长度排序就可以发现规律，式子长度递增，未知数随之递增

换言之，第i个方程中只出现了1-i的数字，不会出现大于i的变量

那么只需要递推解一元一次方程就可以解决了

问题来了：怎么把字符串a[x]转换成列表a中的值呢？

想了许久，正则什么的好像都解决不了问题

查了一下writeup，有一篇是通过eval函数解决的

这才想起来Python自带一个eval函数可以将字符串转成表达式进行求值，那么只需要将方程的'=='用replace函数替换成-()，再将要求的a[

由于式子中仅存在'+a[i]'或'-a[i]'，只需要验证符号后将其去除即可

注册机：

```
#a.length==47&&

#将非方程的第一个式子去除s=[括号内式子]


k=s.split("&&")
k.sort(key=lambda x:len(x))
a=[]

for i,v in enumerate(k):
    v1=v.replace('==','-(')+')'
     v2=v1.replace('a[%d]'%i,'b')
    if(v2[v2.index('b')-1]=='-'):
        v3=v2.replace('-b','')
        a.append(eval(v3))
    else:
        if(v2[v2.index('b')-1]=='+'):
            v3=v2.replace('+b','')
            a.append(-1*eval(v3))
        else:
            v3=v2.replace('b','')
            a.append(-1*eval(v3))
for i in a:
    print(chr(i),end='')
```

其中，倒数第三个式子刚开始总是求出来负值，很明显出错了。对照调试了半天也没发现错误，因为明明倒数第四个式子也是a[i]开头的，最后仔细看了一下发现由于a[i]被去掉，原本的减a[9]%a[42]变为了负a[9]%a[42]，运算优先级被改变了。因为只有这一个地方有问题，所以我直接修改了原方程，在上面加了括号；如果有多处出现这样的问题的话可以对index('b')+1进行检测；

另外，看的writeup上有一个思路很新奇，特记录于此：

我是将a[i]连带符号提出，求表达式的值

由于目的是将a[i]分离，除了我的方法，该writeup是将a[i]转成虚数分离，最后结果a[i]就是虚数i的值，即a[i]= -（实数部/虚数部）

C.明日计划

CrackMe(24)