# 2021 绿城杯 re wp

原创

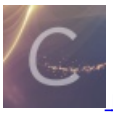
yaoxixixi 于 2021-09-29 22:03:08 发布 216 收藏 1

分类专栏： re 文章标签： 算法 python 概率论



本文链接：https://blog.csdn.net/qq_45739995/article/details/120557091







## 这里写目录标题



这次绿城杯又被大师傅带飞了，我又明明白白的躺了一次，比赛一共3个re +一个安卓

我只做了两个，其他被师傅们秒了

## easy_re

(二血)

ida打开 发现是rc4算法

里面有一点小改变 中间异或了0x37

```
strcpy(key, "tallmewhy");
memset(&key[10], 0, 0xF6u);
v19[0] = xmmword_4021B0;
v19[1] = xmmword_4021C0;
v20 = 0x54E6D699;
v21 = 340807546;
v22 = -4891;
puts("Hello, this is my world.If you want flag, give me something I like.");
sub_401010("\n", v16);
memset(v24, 0, sizeof(v24));
gets(v24);
v3 = strlen(v24);
v4 = strlen(key);
memset(b, 0, sizeof(b));
for ( i = 0; i < 256; ++i )
{
  s[i] = i;
  b[i] = key[i % v4];
}
ii = 0;
jj = 0;
do
{
  q = s[ii];
  jj = (jj + b[ii] + q) % 256;
  s[ii++] = s[jj];
  s[jj] = q ^ 0x37;
}
while ( ii < 256 );
sub_401010("\n\n", v17);
```

key

密文

小改变

```c
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
int b[300];
int s[300];
char key[10]="tallmewhy";//(变)
char data[1000]={
  0xF5, 0x8C, 0x8D, 0xE4, 0x9F, 0xA5, 0x28, 0x65, 0x30, 0xF4,
  0xEB, 0xD3, 0x24, 0xA9, 0x91, 0x1A, 0x6F, 0xD4, 0x6A, 0xD7,
  0x0B, 0x8D, 0xE8, 0xB8, 0x83, 0x4A, 0x5A, 0x6E, 0xBE, 0xCB,
  0xF4, 0x4B, 0x99, 0xD6, 0xE6, 0x54, 0x7A, 0x4F, 0x50, 0x14,
  0xE5, 0xEC
};//(变)
int main(){
 int j,q,n;
 for(int i=0;i<256;i++){
  b[i]=key[i%9];  //8是密钥的长度 （变）
  s[i]=i;
 }
 for(int i=0;i<256;i++){
  j=(j+s[i]+b[i])%256;
  q=s[i];
  s[i]=s[j];
  s[j]=q^0x37;
 }
 for(int i=0;i<256;i++){
  printf("%d ",s[i]);
 }
 printf("\n");
 int i=0,t;
 j=0;
 for(int w=0;w<42;w++){  //32是data的长度 （变）

  i=(i+1)%256;
  j=(j+s[i])%256;

  q=s[i];
  s[i]=s[j];
  s[j]=q; //交换

  t=(s[i]+s[j])%256;
  data[i-1]^=s[t];//s[t]是最后的密钥
 }
 puts(data);


 return 0;
}
```

## Helloworld

代码含有大量花指令，去除后可以定位到主要加密代码

```
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 48, 4) = 136;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 52, 4) = 10;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 56, 4) = 130;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 60, 4) = 185;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 64, 4) = 49;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 68, 4) = 141;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 72, 4) = 10;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 76, 4) = 253;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 80, 4) = 201;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 84, 4) = 199;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 88, 4) = 127;
```

```
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 92, 4) = 185;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 96, 4) = 17;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 100, 4) = 78;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 104, 4) = 185;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 108, 4) = 232;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 112, 4) = 141;
*sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 21, a2 - 124, 116, 4) = 87;
*(a2 - 196) = sub_34C0(a2 - 176);
v2 = 0;
if ( *(a2 - 196) > 0 )
{
  *(a2 - 200) = a2 - 176;
  do
  {
    v6 = *(a2 - 196);
    v3 = sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 24, *(a2 - 200), v2, 0);
    (loc_330)(v3, v6);
    *(a2 - 188) = *sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 26, a2 - 176, v2, 1);
    if ( *sub_2450("C:/WindRiver/workspace/helloworld/helloworld.c", 26, a2 - 124, 4 * v2, 4) == *(a2 - 188) )
      ++*(a2 - 192);
    ++v2;
  }
```



这里有flag加密后的比较数据，加密算法都在loc_330里面
因为花指令没有除干净（太菜了，不会除了），可以配合着汇编代码看

```
text:00000340 018                cmp     eax, 0
text:00000343 018                jnz     short loc_363
text:00000345 018                jz      short loc_35C
text:00000347 018                jnz     short loc_35C
text:00000349 018                lea     eax, [esp-4]
text:0000034D 018                xor     eax, 22h
text:00000350 018                cmp     eax, [esp]
text:00000353 018                jz      short loc_35F
text:00000355 018                call    sub_379
text:0000035A 018                jz      short loc_360
text:0000035C
text:0000035C     loc_35C:                               ; CODE XREF: sub_330+15↑j
text:0000035C                                            ; sub_330+17↑j
text:0000035C 018                nop                     ; Keypatch modified this from:
text:0000035C                                            ;   db 0B8h
text:0000035D 018                add     [eax], eax
text:0000035F
text:0000035F     loc_35F:                               ; CODE XREF: sub_330+23↑j
text:0000035F 018                nop                     ; Keypatch filled range [0x35F:0x360]
text:0000035F                                            ;   db 0
text:0000035F                                            ;   db 0
text:00000360
text:00000360     loc_360:                               ; CODE XREF: sub_330+2A↑j
text:00000360 018                nop
text:00000361 018                jmp     short loc_3C8
text:00000363     ; ---------------------------------------------------------------------------
text:00000363
text:00000363     loc_363:                               ; CODE XREF: sub_330+13↑j
text:00000363 018                push    1
text:00000365 01C                push    0
text:00000367 020                push    [ebp+arg_0]
text:0000036A 024                push    0Ah
text:0000036C 028                push    offset aCWindriverWork ; "C:/WindRiver/workspace/hell
text:00000371 02C                call    sub_2450
text:00000376 02C                add     esp, 14h
text:00000376     sub_330        endp ; sp-analysis failed
text:00000376
```



这里会对flag异或0x22

```
  int v5; // eax
  int v7; // [esp-4h] [ebp-8h] BYREF
  int v8; // [esp+0h] [ebp-4h]

  *(a2 - 12) = a1;
  **(a2 - 12) ^= 0x22u;
  *(a2 - 16) = (sub_2450)("C:/WindRiver/workspace/helloworld/helloworld.c", 11, *(a2 + 8), 0, 1, v8);
  v2 = *(a2 - 16);
  v3 = *v2 == 0xFD;
  *v2 += 3;
  if ( v3 || !v3 )
    goto LABEL_5;
  v4 = &v7 ^ 0x22;
  if ( (&v7 ^ 0x22) == v8 )
  {
LABEL_6:
    v5 = v4 - 1;
    return (loc_330)(*(a2 + 8), v5);
  }
  v5 = ((&loc_3D3 + 2))();
  if ( !v3 )
  {
LABEL_5:
    v4 = *(a2 + 12);
    goto LABEL_6;
  }
  return (loc_330)(*(a2 + 8), v5);
}
```

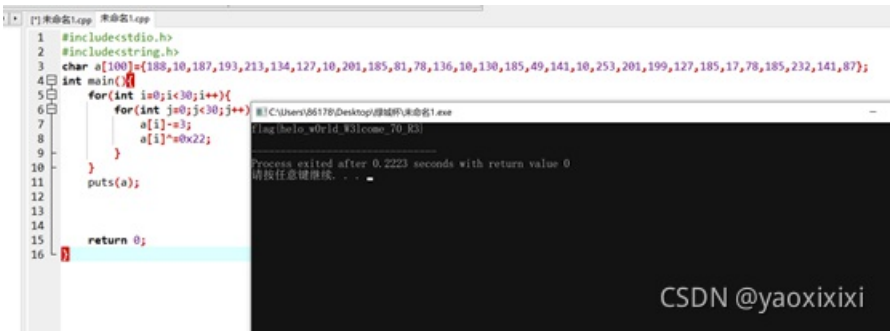


下面的sub_379里会对flag加3操作
写脚本

```
#include<stdio.h>
#include<string.h>
char a[100]={188,10,187,193,213,134,127,10,201,185,81,78,136,10,130,185,49,141,10,253,201,199,127,185,17,78,185,
232,141,87};
int main(){
 for(int i=0;i<30;i++){
  for(int j=0;j<30;j++){
   a[i]-=3;
   a[i]^=0x22;
  }
 }
 puts(a);



 return 0;
}
```




到时候要再复习一下花指令，老弄得不漂亮

## 抛石机

null-404师傅解出来的，我直接copywp（嘻嘻
上课前大致浏览了一下题目，难点在于小数和16进制的转化，以前在buu上做过一道这种类似的题

□先打开ida，定位主函数：
□先进□逻辑判断，判断是否是flag{xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx}的格式，x的取值为1-9
或者a-f
判断如下：

```
__int64 __fastcall sub_1155(char a1)
{
  unsigned int v2; // [rsp+10h] [rbp-4h]

  v2 = -1;
  if ( a1 <= 96 || a1 > 102 )
  {
    if ( a1 > 47 && a1 <= 57 )
      v2 = a1 - 48;
  }
  else
  {
```

```
    v2 = a1 - 87;
  }
  return v2;
}
```

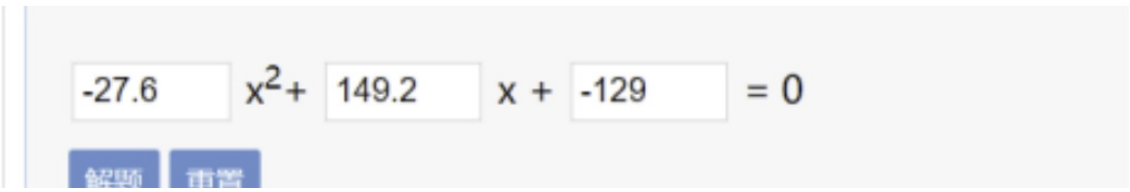


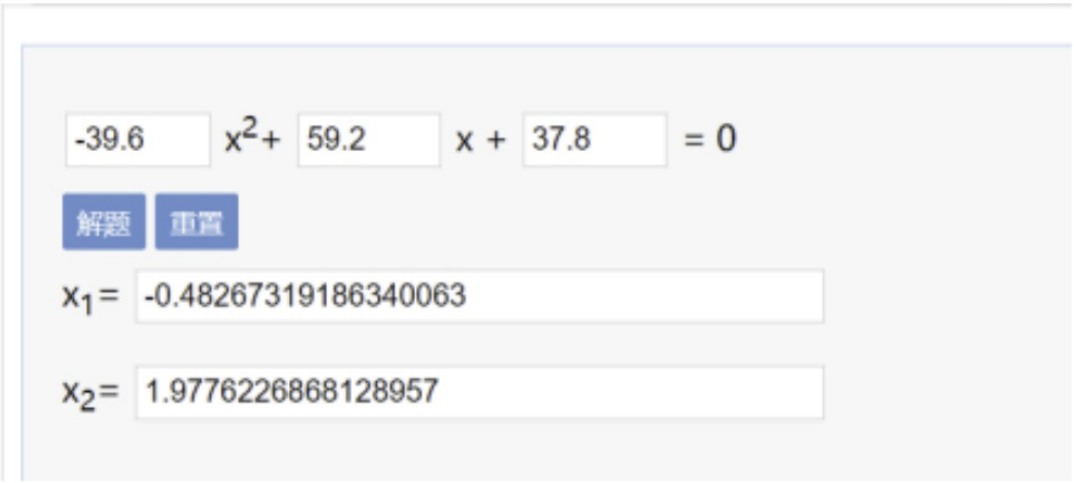
```
__isoc99_scanf("%43s", &v27);
v15 = 'xxx{galf';
v16 = 'xx-xxxxx';
v8 = 'xxx-xxxx';
v17 = '-xxxx-xx';
v18 = 'xxx-xxxx';
v19 = 'xxxxxxxx';
v20 = '}x';
v21 = 0;
for ( i = 0; i <= 42; ++i )
{
  if ( *((_BYTE *)&v15 + i) == 120 )
  {
    v4 = (unsigned int)*((char *)&v27 + i);
    if ( (signed int)sub_1155(*((_BYTE *)&v27 + i)) < 0 )
    {
      puts("you lost!");
      exit(1);
    }
    v8 = *((unsigned __int8 *)&v27 + i);
    *((_BYTE *)&v22 + v37++) = v8;
  }
  else
  {
    v8 = *((unsigned __int8 *)&v27 + i);
    if ( (_BYTE)v8 != *((_BYTE *)&v15 + i) )
    {
      puts("you lost!");
      exit(1);
    }
  }
}
```



接着，将输□的值，不包括flag{-}，转换为double类型，转换□式为每8个字符的□进制变为□个double。然后进□解□程：

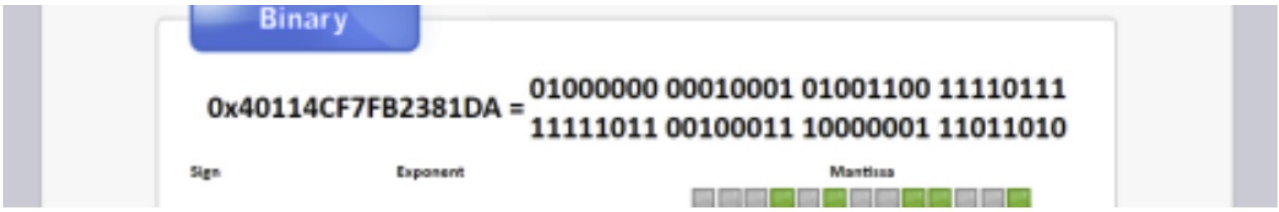
```
v4 = 149.2 * *(double *)&doule1 + *(double *)&doule1 * -27.6 * *(double *)&doule1 - 129.0;
v3 = 149.2 * *(double *)&doule2 + *(double *)&doule2 * -27.6 * *(double *)&doule2 - 129.0;
v2 = *(double *)&doule3 * -39.6 * *(double *)&doule3 + 59.2 * *(double *)&doule3 + 37.8;
v1 = *(double *)&doule4 * -39.6 * *(double *)&doule4 + 59.2 * *(double *)&doule4 + 37.8;
```






解出后进□□进制转化：








转化后取前8位，□□写转换：3FF14A45 40114CF7 BFDEE41E 3FFFA457
然后进□倒叙输出：454AF13F F74C1140 1EE4DEBF 57A4FF3F
然后进□调式，根据出错的位置对精度进□调整，调整后得到：
454AF13F F64C1140 1EE4DEBF 58A4FF3F
然后根据格式调整为

然后根据格式调整为：

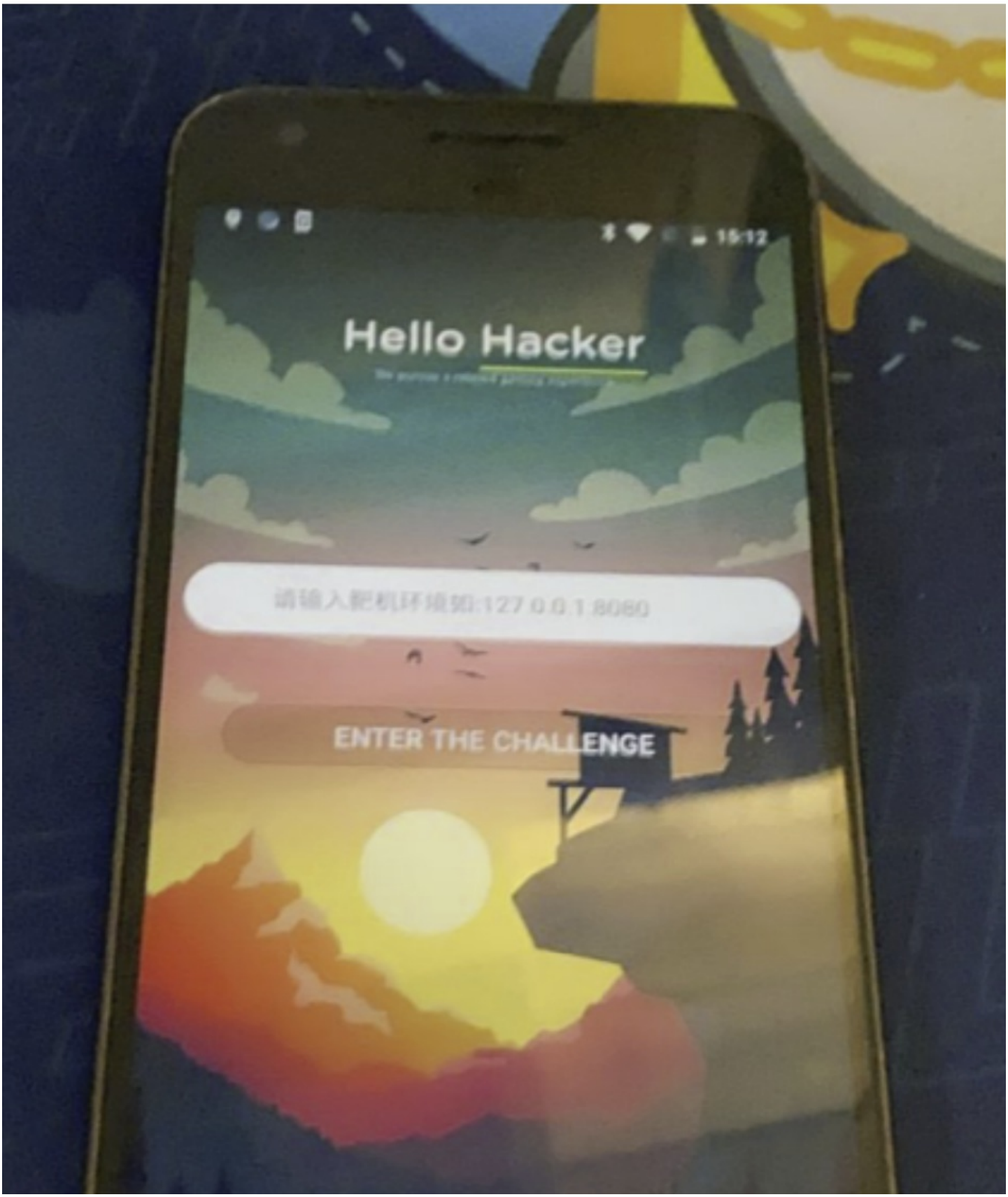
flag{454af13f-f84c-1140-1ee4-debf58a4ff3f}

## clockin

安卓题 由0xK4ws解出 （看到真机羡慕了！！）下面是师傅的wp

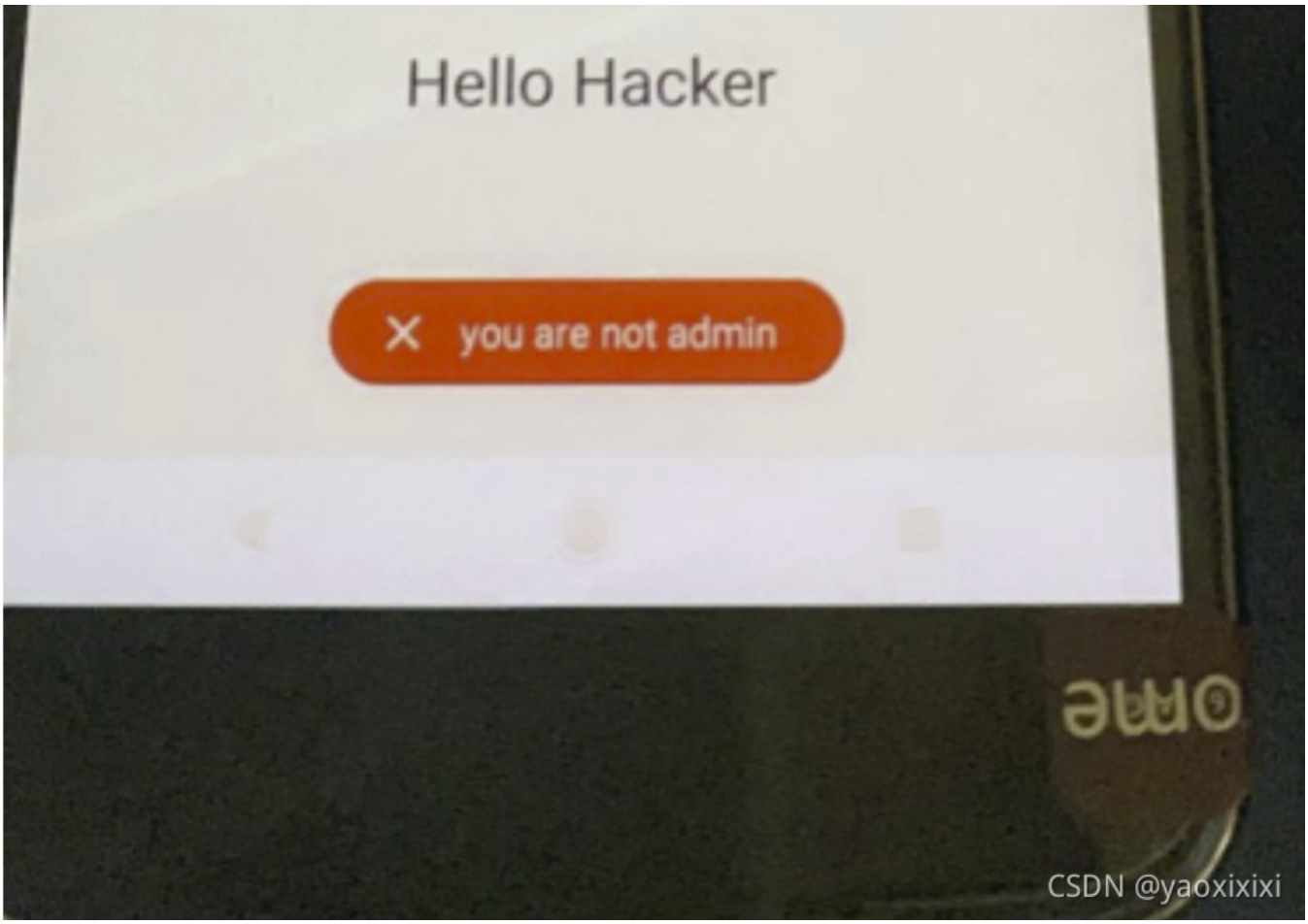
```
adb install -r -t myApk.apk
```

□先安装程序 □以上命令安装程序□可安装成功



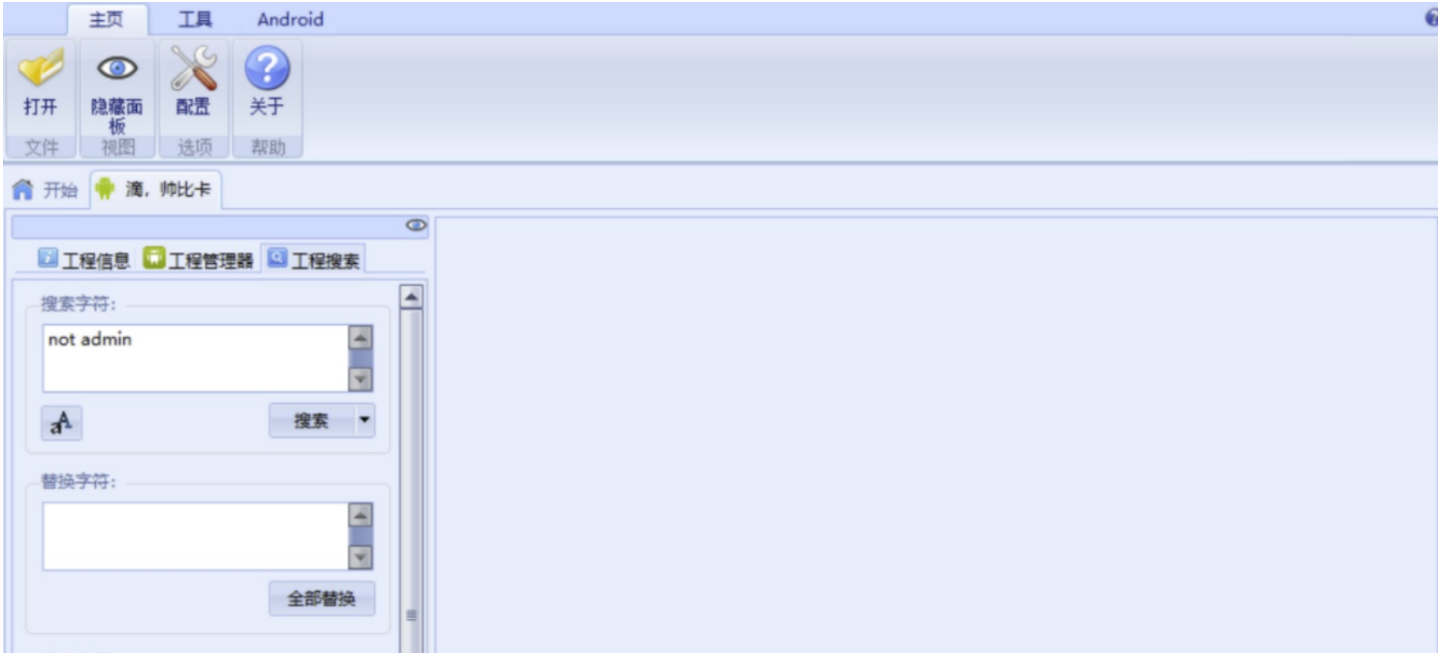

CSDN @yaoxixixi

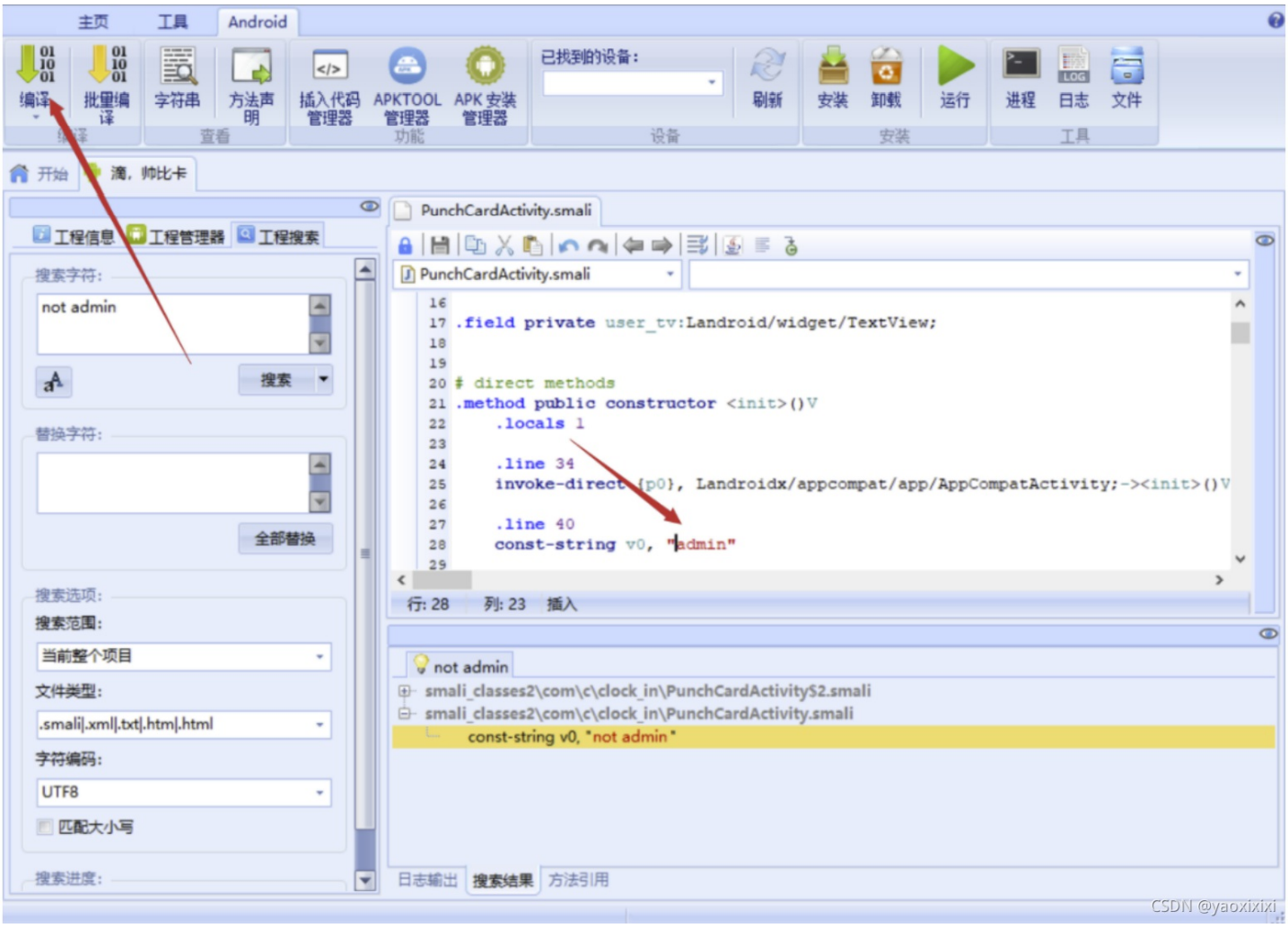

Punch card to get flag
滴，帅比卡



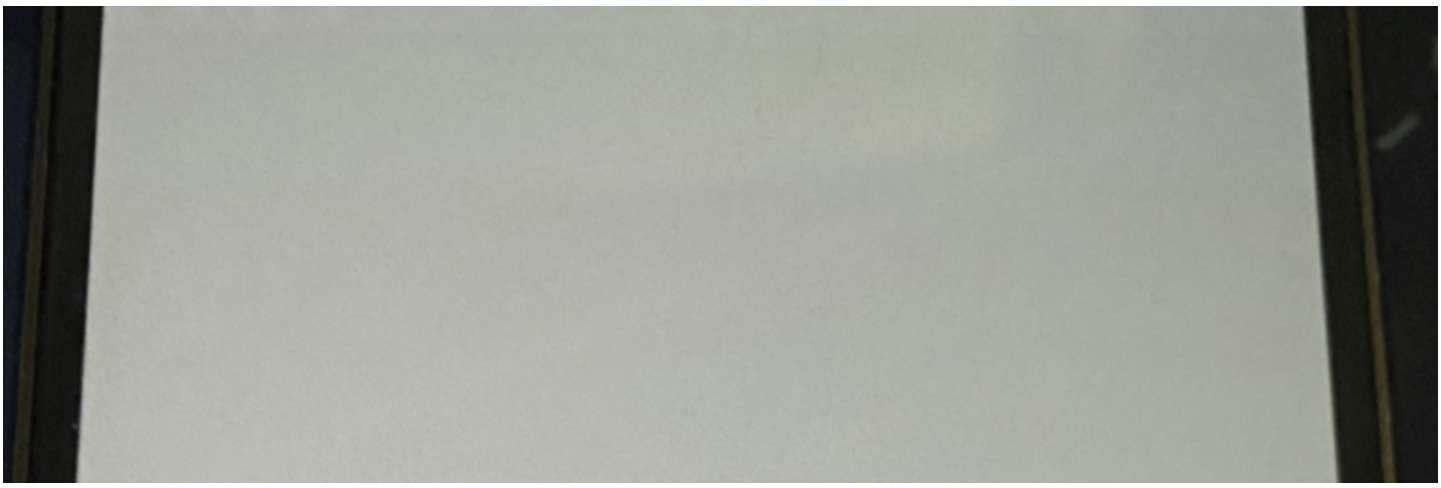

```
private String ip_port;
private String permission = "not admin";
private ImageView punch_card_iv;
```

在Android killer上搜索字符串not admin
把not admin改成admin重新打包




然后启动该APP 输□地址和端□ 点击打卡拿到flag




## 总结

要学习的东西还是很多，做题的速度也应该提高，多学习学习别人的wp