#### BUU-RE

## 1.刮开有奖

BOOL \_\_stdcall DialogFunc(HWND hDlg, UINT a2, WPARAM a3, LPARAM a4)
  
{
  
 const char \*v4; // esi
  
 const char \*v5; // edi
  
 int a[0]; // [esp+8h] [ebp-20030h]
  
 int a[1]; // [esp+Ch] [ebp-2002Ch]
  
 int a[2]; // [esp+10h] [ebp-20028h]
  
 int a[3]; // [esp+14h] [ebp-20024h]
  
 int a[4]; // [esp+18h] [ebp-20020h]
  
 int a[5]; // [esp+1Ch] [ebp-2001Ch]
  
 int a[6]; // [esp+20h] [ebp-20018h]
  
 int a[7]; // [esp+24h] [ebp-20014h]
  
 int a[8]; // [esp+28h] [ebp-20010h]
  
 int a[9]; // [esp+2Ch] [ebp-2000Ch]
  
 int a[10]; // [esp+30h] [ebp-20008h]
  
 CHAR string[0]; // [esp+34h] [ebp-20004h]
  
 char string[1]; // [esp+35h] [ebp-20003h]
  
 char string[2]; // [esp+36h] [ebp-20002h]
  
 char string[3]; // [esp+37h] [ebp-20001h]
  
 char string[4]; // [esp+38h] [ebp-20000h]
  
 char string[5]; // [esp+39h] [ebp-1FFFFh]
  
 char string[6]; // [esp+3Ah] [ebp-1FFFEh]
  
 char string[7]; // [esp+3Bh] [ebp-1FFFDh]
  
 char v26; // [esp+10034h] [ebp-10004h]
  
 char v27; // [esp+10035h] [ebp-10003h]
  
 char v28; // [esp+10036h] [ebp-10002h]
  
   
 if ( a2 == 272 )
  
 return 1;
  
 if ( a2 != 273 ) // a2 = 273
  
 return 0;
  
 if ( (\_WORD)a3 == 1001 )
  
 {
  
 memset(&string[0], 0, 0xFFFFu);
  
 GetDlgItemTextA(hDlg, 1000, &string[0], 0xFFFF);
  
 if ( strlen(&string[0]) == 8 )
  
 {
  
 a[0] = 90; // Z
  
 a[1] = 74; // J
  
 a[2] = 83; // S
  
 a[3] = 69; // E
  
 a[4] = 67; // C
  
 a[5] = 97; // a
  
 a[6] = 78; // N
  
 a[7] = 72; // H
  
 a[8] = 51; // 3
  
 a[9] = 110; // n
  
 a[10] = 103; // g
  
 sub\_4010F0((int)&a[0], 0, 10); // 排序
  
 // 3,C,E,H,J,N,S,Z,a,g
  
 memset(&v26, 0, 0xFFFFu);
  
 v26 = string[5];
  
 v28 = string[7];
  
 v27 = string[6];
  
 v4 = sub\_401000((int)&v26, strlen(&v26)); // base64
  
 memset(&v26, 0, 0xFFFFu);
  
 v27 = string[3];
  
 v26 = string[2];
  
 v28 = string[4];
  
 v5 = sub\_401000((int)&v26, strlen(&v26));
  
 if ( string[0] == a[0] + 34 // 51+34=85 U
  
 && string[1] == a[4] // J
  
 && 4 \* string[2] - 141 == 3 \* a[2] // W
  
 && string[3] / 4 == 2 \* (a[7] / 9) // P
  
 && !strcmp(v4, "ak1w") // WP1
  
 && !strcmp( // jMp
  
 v5,
  
 "V1Ax") )
  
 {
  
 MessageBoxA(hDlg, "U g3t 1T!", "@\_@", 0);
  
 }
  
 }
  
 return 0;
  
 }
  
 if ( (\_WORD)a3 != 1 && (\_WORD)a3 != 2 )
  
 return 0;
  
 EndDialog(hDlg, (unsigned \_\_int16)a3);
  
 return 1;
  
}

sub\_4010F0()：//排序

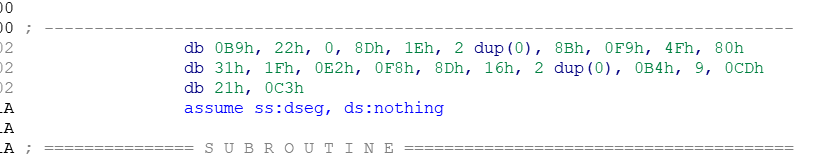
int \_\_cdecl sub\_4010F0(int a1, int a2, int a3)  
{  
 int result; // eax  
 int i; // esi  
 int v5; // ecx  
 int v6; // edx  
  
 result = a3;  
 for ( i = a2; i <= a3; a2 = i )  
 {  
 v5 = 4 \* i;  
 v6 = \*(\_DWORD \*)(4 \* i + a1);  
 if ( a2 < result && i < result )  
 {  
 do  
 {  
 if ( v6 > \*(\_DWORD \*)(a1 + 4 \* result) )  
 {  
 if ( i >= result )  
 break;  
 ++i;  
 \*(\_DWORD \*)(v5 + a1) = \*(\_DWORD \*)(a1 + 4 \* result);  
 if ( i >= result )  
 break;  
 while ( \*(\_DWORD \*)(a1 + 4 \* i) <= v6 )  
 {  
 if ( ++i >= result )  
 goto LABEL\_13;  
 }  
 if ( i >= result )  
 break;  
 v5 = 4 \* i;  
 \*(\_DWORD \*)(a1 + 4 \* result) = \*(\_DWORD \*)(4 \* i + a1);  
 }  
 --result;  
 }  
 while ( i < result );  
 }  
LABEL\_13:  
 \*(\_DWORD \*)(a1 + 4 \* result) = v6;  
 sub\_4010F0(a1, a2, i - 1);  
 result = a3;  
 ++i;  
 }  
 return result;  
}

sub\_401000()://base64

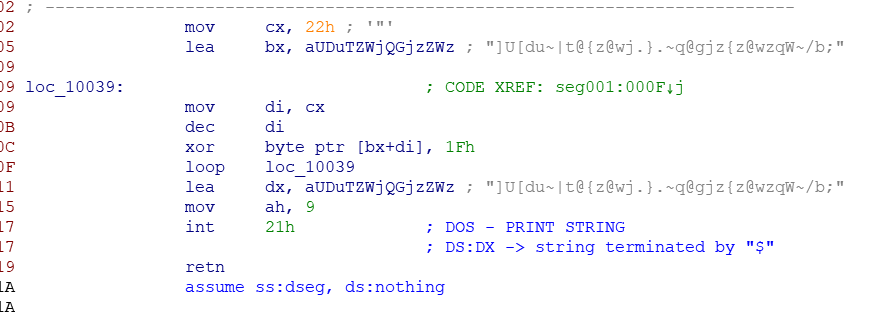
\_BYTE \*\_\_cdecl sub\_401000(int a1, int a2)  
{  
 int v2; // eax  
 int v3; // esi  
 size\_t v4; // ebx  
 \_BYTE \*v5; // eax  
 \_BYTE \*v6; // edi  
 int v7; // eax  
 \_BYTE \*v8; // ebx  
 int v9; // edi  
 signed int v10; // edx  
 int v11; // edi  
 signed int v12; // eax  
 signed int v13; // esi  
 \_BYTE \*result; // eax  
 \_BYTE \*v15; // [esp+Ch] [ebp-10h]  
 \_BYTE \*v16; // [esp+10h] [ebp-Ch]  
 int v17; // [esp+14h] [ebp-8h]  
 int v18; // [esp+18h] [ebp-4h]  
  
 v2 = a2 / 3;  
 v3 = 0;  
 if ( a2 % 3 > 0 )  
 ++v2;  
 v4 = 4 \* v2 + 1;  
 v5 = malloc(v4);  
 v6 = v5;  
 v15 = v5;  
 if ( !v5 )  
 exit(0);  
 memset(v5, 0, v4);  
 v7 = a2;  
 v8 = v6;  
 v16 = v6;  
 if ( a2 > 0 )  
 {  
 while ( 1 )  
 {  
 v9 = 0;  
 v10 = 0;  
 v18 = 0;  
 do  
 {  
 if ( v3 >= v7 )  
 break;  
 ++v10;  
 v9 = \*(unsigned \_\_int8 \*)(v3++ + a1) | (v9 << 8);  
 }  
 while ( v10 < 3 );  
 v11 = v9 << 8 \* (3 - v10);  
 v12 = 0;  
 v17 = v3;  
 v13 = 18;  
 do  
 {  
 if ( v10 >= v12 )  
 {  
 \*((\_BYTE \*)&v18 + v12) = (v11 >> v13) & 0x3F;  
 v8 = v16;  
 }  
 else  
 {  
 \*((\_BYTE \*)&v18 + v12) = 64;  
 }  
 \*v8++ = byte\_407830[\*((char \*)&v18 + v12)];  
 v13 -= 6;  
 ++v12;  
 v16 = v8;  
 }  
 while ( v13 > -6 );  
 v3 = v17;  
 if ( v17 >= a2 )  
 break;  
 v7 = a2;  
 }  
 v6 = v15;  
 }  
 result = v6;  
 \*v8 = 0;  
 return result;  
}

得到flag

## 2.[BJDCTF 2nd]8086



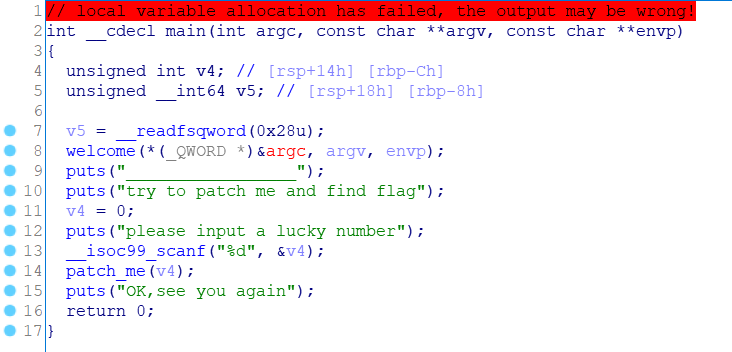
选中，c，force，强制汇编



dec为自减

key = ']U[du~|t@{z@wj.}.~q@gjz{z@wzqW~/b'  
flag=''  
for i in key:  
 a=ord(i)^0x1f  
 flag+=chr(a)  
print flag

## 3.[GXYCTF2019]luck\_guy



patch\_me():

int \_\_fastcall patch\_me(int a1)  
{  
 int result; // eax  
  
 if ( a1 % 2 == 1 )  
 result = puts("just finished");  
 else  
 result = get\_flag();  
 return result;  
}

get\_flag():

unsigned \_\_int64 get\_flag()  
{  
 unsigned int v0; // eax  
 char v1; // al  
 signed int i; // [rsp+4h] [rbp-3Ch]  
 signed int j; // [rsp+8h] [rbp-38h]  
 \_\_int64 s; // [rsp+10h] [rbp-30h]  
 char v6; // [rsp+18h] [rbp-28h]  
 unsigned \_\_int64 v7; // [rsp+38h] [rbp-8h]  
  
 v7 = \_\_readfsqword(0x28u);  
 v0 = time(0LL);  
 srand(v0);  
 for ( i = 0; i <= 4; ++i )  
 {  
 switch ( rand() % 200 )  
 {  
 case 1:  
 puts("OK, it's flag:");  
 memset(&s, 0, 0x28uLL);  
 strcat((char \*)&s, f1);  
 strcat((char \*)&s, &f2);  
 printf("%s", &s);  
 break;  
 case 2:  
 printf("Solar not like you");  
 break;  
 case 3:  
 printf("Solar want a girlfriend");  
 break;  
 case 4:  
 v6 = 0;  
 s = 'fo`guci';  
 strcat(&f2, (const char \*)&s);  
 break;  
 case 5:  
 for ( j = 0; j <= 7; ++j )  
 {  
 if ( j % 2 == 1 )  
 v1 = \*(&f2 + j) - 2;  
 else  
 v1 = \*(&f2 + j) - 1;  
 \*(&f2 + j) = v1;  
 }  
 break;  
 default:  
 puts("emmm,you can't find flag 23333");  
 break;  
 }  
 }  
 return \_\_readfsqword(0x28u) ^ v7;  
}

执行顺序：451

f1 db 'GXY{do\_not\_',0

flag = 'GXY{do\_not\_'  
f2 = [0x7F, 0x66, 0x6F, 0x60, 0x67, 0x75, 0x63, 0x69][::-1]  
s = ''  
for i in range(8):  
 if i % 2 == 1:  
 s = chr(int(f2[i]) - 2)  
 else:  
 s = chr(int(f2[i]) - 1)  
 flag += s  
print(flag)

GXY{do\_not\_hate\_me}

## 4.[BJDCTF2020]JustRE

BOOL \_\_stdcall DialogFunc(HWND hWnd, UINT a2, WPARAM a3, LPARAM a4)  
{  
 CHAR String; // [esp+0h] [ebp-64h]  
  
 if ( a2 != 272 )  
 {  
 if ( a2 != 273 )  
 return 0;  
 if ( (\_WORD)a3 != 1 && (\_WORD)a3 != 2 )  
 {  
 sprintf(&String, &byte\_40704C, ++dword\_4099F0);  
 if ( dword\_4099F0 == 19999 )  
 {  
 sprintf(&String, aBjdDD2069a4579, 19999, 0);  
 SetWindowTextA(hWnd, &String);  
 return 0;  
 }  
 SetWindowTextA(hWnd, &String);  
 return 0;  
 }  
 EndDialog(hWnd, (unsigned \_\_int16)a3);  
 }  
 return 1;  
}

aBjdDD2069a4579 db ' BJD{%d%d2069a45792d233ac}',0

BJD{1999902069a45792d233ac}

## 5.CrackRTF

拖入IDA

int main\_0()
  
{
  
 DWORD v0; // eax
  
 DWORD v1; // eax
  
 CHAR String; // [esp+4Ch] [ebp-310h]
  
 int v4; // [esp+150h] [ebp-20Ch]
  
 CHAR String1; // [esp+154h] [ebp-208h]
  
 BYTE pbData; // [esp+258h] [ebp-104h]
  
   
 memset(&pbData, 0, 0x104u);
  
 memset(&String1, 0, 0x104u);
  
 v4 = 0;
  
 printf("pls input the first passwd(1): ");
  
 scanf("%s", &pbData);
  
 if ( strlen((const char \*)&pbData) != 6 )
  
 {
  
 printf("Must be 6 characters!\n");
  
 ExitProcess(0);
  
 }
  
 v4 = atoi((const char \*)&pbData);
  
 if ( v4 < 100000 )
  
 ExitProcess(0);
  
 strcat((char \*)&pbData, "@DBApp");
  
 v0 = strlen((const char \*)&pbData);
  
 sub\_40100A(&pbData, v0, &String1);
  
 if ( !\_strcmpi(&String1, "6E32D0943418C2C33385BC35A1470250DD8923A9") )
  
 {
  
 printf("continue...\n\n");
  
 printf("pls input the first passwd(2): ");
  
 memset(&String, 0, 0x104u);
  
 scanf("%s", &String);
  
 if ( strlen(&String) != 6 )
  
 {
  
 printf("Must be 6 characters!\n");
  
 ExitProcess(0);
  
 }
  
 strcat(&String, (const char \*)&pbData);
  
 memset(&String1, 0, 0x104u);
  
 v1 = strlen(&String);
  
 sub\_401019((BYTE \*)&String, v1, &String1);
  
 if ( !\_strcmpi("27019e688a4e62a649fd99cadaafdb4e", &String1) )
  
 {
  
 if ( !(unsigned \_\_int8)sub\_40100F(&String) )
  
 {
  
 printf("Error!!\n");
  
 ExitProcess(0);
  
 }
  
 printf("bye ~~\n");
  
 }
  
 }
  
 return 0;
  
}

输入两次，每次输入都为6位，如果错误就退出

atoi函数：把字符串转换成整型数的一个函数（前面有几个数字就输出那几个，首位字符及之后的不输出）

再之后与”@DAAPP“字符串拼接，调用sub*40100A函数（sub*401230）:

int \_\_cdecl sub\_401230(BYTE \*pbData, DWORD dwDataLen, LPSTR lpString1)
  
{
  
 int result; // eax
  
 DWORD i; // [esp+4Ch] [ebp-28h]
  
 CHAR String2; // [esp+50h] [ebp-24h]
  
 BYTE v6[20]; // [esp+54h] [ebp-20h]
  
 DWORD pdwDataLen; // [esp+68h] [ebp-Ch]
  
 HCRYPTHASH phHash; // [esp+6Ch] [ebp-8h]
  
 HCRYPTPROV phProv; // [esp+70h] [ebp-4h]
  
 if ( !CryptAcquireContextA(&phProv, 0, 0, 1u, 0xF0000000) )
  
 return 0;
  
 if ( CryptCreateHash(phProv, 0x8004u, 0, 0, &phHash) )
  
 {
  
 if ( CryptHashData(phHash, pbData, dwDataLen, 0) )
  
 {
  
 CryptGetHashParam(phHash, 2u, v6, &pdwDataLen, 0);
  
 \*lpString1 = 0;
  
 for ( i = 0; i < pdwDataLen; ++i )
  
 {
  
 wsprintfA(&String2, "%02X", v6[i]);
  
 lstrcatA(lpString1, &String2);
  
 }
  
 CryptDestroyHash(phHash);
  
 CryptReleaseContext(phProv, 0);
  
 result = 1;
  
 }
  
 else
  
 {
  
 CryptDestroyHash(phHash);
  
 CryptReleaseContext(phProv, 0);
  
 result = 0;
  
 }
  
 }
  
 else
  
 {
  
 CryptReleaseContext(phProv, 0);
  
 result = 0;
  
 }
  
 return result;
  
}

百度后发现是哈希加密，0x8004u查表后发现是sha1加密。MD5解密后为123321@DBApp。

第二次输入一样，0x8003u查表后为MD5，解密为~!3a@0123321@DBApp。

输入两个密码后会有一个文件，里面就有falg。

但是看了其他大佬的wp后发现这么做有些侥幸，他们是这样做的：

首先第一个密码写脚本：

import hashlib
  
   
flag = "@DBApp"
  
   
for i in range(100000,999999):
  
 s = str(i)+flag
  
 x = hashlib.sha1(s.encode())
  
 cnt = x.hexdigest()
  
 if "6e32d0943418c2c" in cnt:
  
 print(cnt)
  
 print(str(i)+flag)

接下来的第二个密码不知道为啥我是somd5解密然后解出来了，但是大佬们是这么做的：

后面的sub*40100F函数(sub*4014D0);

char \_\_cdecl sub\_4014D0(LPCSTR lpString)
  
{
  
 LPCVOID lpBuffer; // [esp+50h] [ebp-1Ch]
  
 DWORD NumberOfBytesWritten; // [esp+58h] [ebp-14h]
  
 DWORD nNumberOfBytesToWrite; // [esp+5Ch] [ebp-10h]
  
 HGLOBAL hResData; // [esp+60h] [ebp-Ch]
  
 HRSRC hResInfo; // [esp+64h] [ebp-8h]
  
 HANDLE hFile; // [esp+68h] [ebp-4h]
  
   
 hFile = 0;
  
 hResData = 0;
  
 nNumberOfBytesToWrite = 0;
  
 NumberOfBytesWritten = 0;
  
 hResInfo = FindResourceA(0, (LPCSTR)0x65, "AAA");
  
 if ( !hResInfo )
  
 return 0;
  
 nNumberOfBytesToWrite = SizeofResource(0, hResInfo);
  
 hResData = LoadResource(0, hResInfo);
  
 if ( !hResData )
  
 return 0;
  
 lpBuffer = LockResource(hResData);
  
 sub\_401005(lpString, (int)lpBuffer, nNumberOfBytesToWrite);
  
 hFile = CreateFileA("dbapp.rtf", 0x10000000u, 0, 0, 2u, 0x80u, 0);
  
 if ( hFile == (HANDLE)-1 )
  
 return 0;
  
 if ( !WriteFile(hFile, lpBuffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0) )
  
 return 0;
  
 CloseHandle(hFile);
  
 return 1;
  
}

将第二个密码与从AAA取出的数据传入sub*401005函数（sub*401420）：

unsigned int \_\_cdecl sub\_401420(LPCSTR lpString, int a2, int a3)
  
{
  
 unsigned int result; // eax
  
 unsigned int i; // [esp+4Ch] [ebp-Ch]
  
 unsigned int v5; // [esp+54h] [ebp-4h]
  
   
 v5 = lstrlenA(lpString);
  
 for ( i = 0; ; ++i )
  
 {
  
 result = i;
  
 if ( i >= a3 )
  
 break;
  
 \*(\_BYTE \*)(i + a2) ^= lpString[i % v5];
  
 }
  
 return result;
  
}

将AAA取出的值与第二次密码连接后的值异或，有一款工具可以直接查看文件中的资源——ResourceHacker

百度rtf文件头前6位：{\rtfl

再取AAA的前6位异或：

s = "{\\rtf1"
  
a = [0x05,0x7D,0x41,0x15,0x26,0x01]
  
flag = ""
  
for i in range(0,len(s)):
  
 x = ord(s[i]) ^ a[i]
  
 flag += chr(x)
  
print(flag)

得到第二次密码：~!3a@0