



Название:	Классика 2
Категория:	Реверс-инжиниринг
Уровень:	Лёгкая
Очки:	150
Описание:	Тут уже немного сложнее, но цель всё ещё проста - ввести верный флаг!
Теги:	Побайтовое шифрование
Автор:	ROP

Прохождение:

Изучаем данный нам файл в IDA.

```

● 13
● 14     _main();
● 15     *Buffer = 0LL;
● 16     v10 = 0LL;
● 17     v11 = 0LL;
● 18     v12 = 0LL;
● 19     v13 = 0LL;
● 20     Str1[0] = -10;
● 21     Str1[1] = 66;
● 22     Str1[2] = -3;
● 23     Str1[3] = -4;
● 24     Str1[4] = -9;
● 25     Str1[5] = 80;
● 26     Str1[6] = 46;
● 27     Str1[7] = -10;
● 28     qmemcpy(v7, "%x&&`VFRgR$i'", 13);
● 29     v7[13] = -4;
● 30     qmemcpy(v8, "R]h|[4", sizeof(v8));
● 31     puts(meme);
● 32     sleep(1LL);
● 33     puts("Hi!");
● 34     puts("Let's see if you know the classics of Codeby!");
● 35     sleep(1LL);
● 36     printf("Answer: ");
● 37     v3 = __acrt_iob_func(1u);
● 38     fflush(v3);
● 39     v4 = __acrt_iob_func(0);
● 40     fgets(Buffer, 40, v4);
● 41     Buffer[strcspn(Buffer, "\n")] = 0;
● 42     sub_1229(Buffer, 28LL);
● 43     if (!strcmp(Str1, Buffer, 0x10uLL) )
● 44     {
● 45         puts("You're right!");
● 46         puts("This is your flag! =D");
● 47     }
● 48     else
● 49     {
● 50         puts("That's not it...");
● 51     }
● 52     return 0;

```

00000AA3 main:42 (4016A3)

Он получает строку и вызывает функцию шифрования. Далее проверяется зашифрованный массив на основе введённой строки с нужным зашифрованным массивом.

В `main` интересна выделенная функция.

```
1 | int64 __fastcall sub_1229(__int64 a1, int a2)
2 | {
3 |     __int64 result; // rax
4 |     unsigned int i; // [rsp+Ch] [rbp-4h]
5 |
6 |     for ( i = 0; ; ++i )
7 |     {
8 |         result = i;
9 |         if ( i >= a2 )
10 |             break;
11 |         *(a1 + i) ^= 0x41u;
12 |         *(a1 + i) += 52;
13 |         *(a1 + i) ^= 0x22u;
14 |         *(a1 + i) -= 30;
15 |     }
16 |     return result;
17 | }
```

Происходит последовательное побайтовое шифрование:

1. XOR с 0x41
2. Сложение с 52
3. XOR с 0x22
4. Вычитание с 30

А зашифрованный флаг находится здесь в функции `main`.

```
.text:000000000040155F mov    [rbp+arg_8], rax
.text:0000000000401563 call   __main
.text:0000000000401568 mov    qword ptr [rbp+Buffer], 0
.text:0000000000401570 mov    [rbp+var_28], 0
.text:0000000000401578 mov    [rbp+var_20], 0
.text:0000000000401580 mov    [rbp+var_18], 0
.text:0000000000401588 mov    [rbp+var_10], 0
.text:0000000000401590 mov    [rbp+Str1], 0F6h
.text:0000000000401594 mov    [rbp+var_4F], 42h ; 'B'
.text:0000000000401598 mov    [rbp+var_4E], 0FDh
.text:000000000040159C mov    [rbp+var_4D], 0FCCh
.text:00000000004015A0 mov    [rbp+var_4C], 0F7h
.text:00000000004015A4 mov    [rbp+var_4B], 50h ; 'P'
.text:00000000004015A8 mov    [rbp+var_4A], 2Eh ; '.'
.text:00000000004015AC mov    [rbp+var_49], 0F6h
.text:00000000004015B0 mov    [rbp+var_48], 25h ; '%'
.text:00000000004015B4 mov    [rbp+var_47], 58h ; 'X'
.text:00000000004015B8 mov    [rbp+var_46], 26h ; '&'
.text:00000000004015BC mov    [rbp+var_45], 26h ; '&'
.text:00000000004015C0 mov    [rbp+var_44], 60h ; ' '
.text:00000000004015C4 mov    [rbp+var_43], 56h ; 'V'
.text:00000000004015C8 mov    [rbp+var_42], 46h ; 'F'
.text:00000000004015CC mov    [rbp+var_41], 52h ; 'R'
.text:00000000004015D0 mov    [rbp+var_40], 67h ; 'g'
.text:00000000004015D4 mov    [rbp+var_3F], 52h ; 'R'
.text:00000000004015D8 mov    [rbp+var_3E], 24h ; '$'
.text:00000000004015DC mov    [rbp+var_3D], 69h ; 'i'
.text:00000000004015E0 mov    [rbp+var_3C], 27h ; '...'
.text:00000000004015E4 mov    [rbp+var_3B], 0FCCh
.text:00000000004015E8 mov    [rbp+var_3A], 52h ; 'R'
.text:00000000004015EC mov    [rbp+var_39], 5Dh ; ']'
.text:00000000004015F0 mov    [rbp+var_38], 68h ; 'h'
.text:00000000004015F4 mov    [rbp+var_37], 5Bh ; '['
.text:00000000004015F8 mov    [rbp+var_36], 5Bh ; '['
.text:00000000004015FC mov    [rbp+var_35], 34h ; '4'
...
.text:0000000000401600 lea    rcx, meme      ; "\n\n"
.text:0000000000401607 call   puts
.text:000000000040160C mov    ecx, 1
.text:0000000000401611 call   sleep
```

Развернём алгоритм и получим флаг.

[https://cyberchef.org/#recipe=From Decimal\('Space',false\)ADD\({'option':'Decimal','string':'30'}\).XOR\('30'\).URLEncode](https://cyberchef.org/#recipe=From Decimal('Space',false)ADD({'option':'Decimal','string':'30'}).XOR('30').URLEncode)