re学习笔记（94）第四届2021美团网络安全高校挑战赛 - Random

原创

Forgo7ten 于 2021-12-12 00:00:00 发布 2381 收藏

分类专栏： ctf小白成长ing # reverse 文章标签： web安全 安全 信息安全 逆向 CTF

本文链接：https://blog.csdn.net/Palmer9/article/details/121873238

动调时候报错，除以0异常

在main函数这里手动制造了一个错误

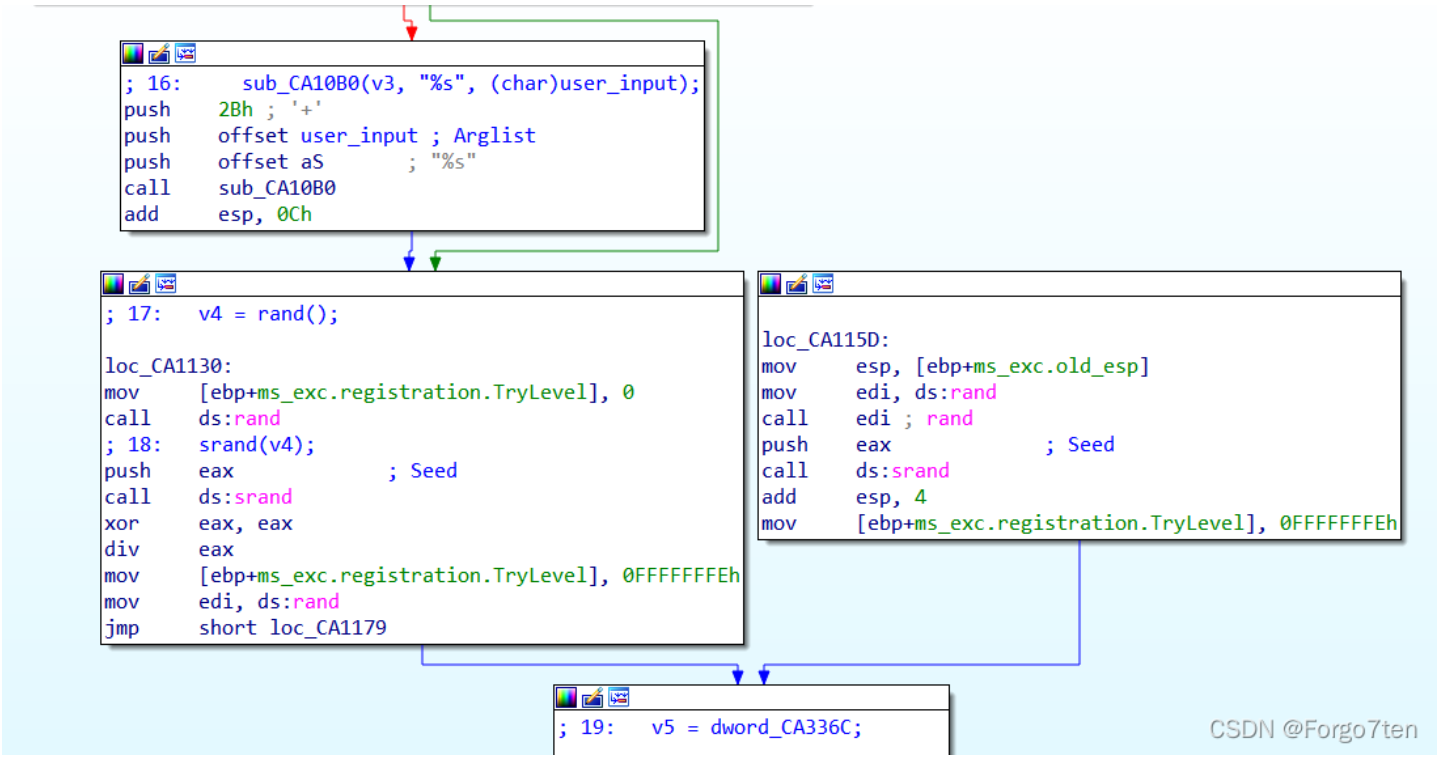
```
.text:00CA1130 ; 17:    v4 = rand();
.text:00CA1130
.text:00CA1130 loc_CA1130:                             ; CODE XREF: _main+3A↑j
.text:00CA1130                 mov     [ebp+ms_exc.registration.TryLevel], 0
.text:00CA1137                 call    ds:rand
.text:00CA113D ; 18:    srand(v4);
.text:00CA113D                 push    eax             ; Seed
.text:00CA113E                 call    ds:srand
.text:00CA1144                 xor     eax, eax
.text:00CA1146                 div     eax
.text:00CA1148                 mov     [ebp+ms_exc.registration.TryLevel], 0FFFFFFFEh
.text:00CA114F                 mov     edi, ds:rand
.text:00CA1155                 jmp     short loc_CA1179
.text:00CA1157 ; ---------------------------------------------------------------------------
```


查看导入表发现有SEH注册的函数

Address	Ordinal	Name	Library
00CA2000		GetSystemTimeAsFileTime	KERNEL32
00CA2004		SetUnhandledExceptionFilter	KERNEL32
00CA2008		GetCurrentProcess	KERNEL32
00CA200C		TerminateProcess	KERNEL32
00CA2010		IsProcessorFeaturePresent	KERNEL32
00CA2014		GetModuleHandleW	KERNEL32
00CA2018		IsDebuggerPresent	KERNEL32
00CA201C		InitializeSListHead	KERNEL32
00CA2020		UnhandledExceptionFilter	KERNEL32
00CA2024		GetCurrentThreadId	KERNEL32
00CA2028		GetCurrentProcessId	KERNEL32

00CA202C QueryPerformanceCounter KERNEL32 CSDN @Forgo7ten

且这里多了个分支



根据交叉引用找到seh注册的异常

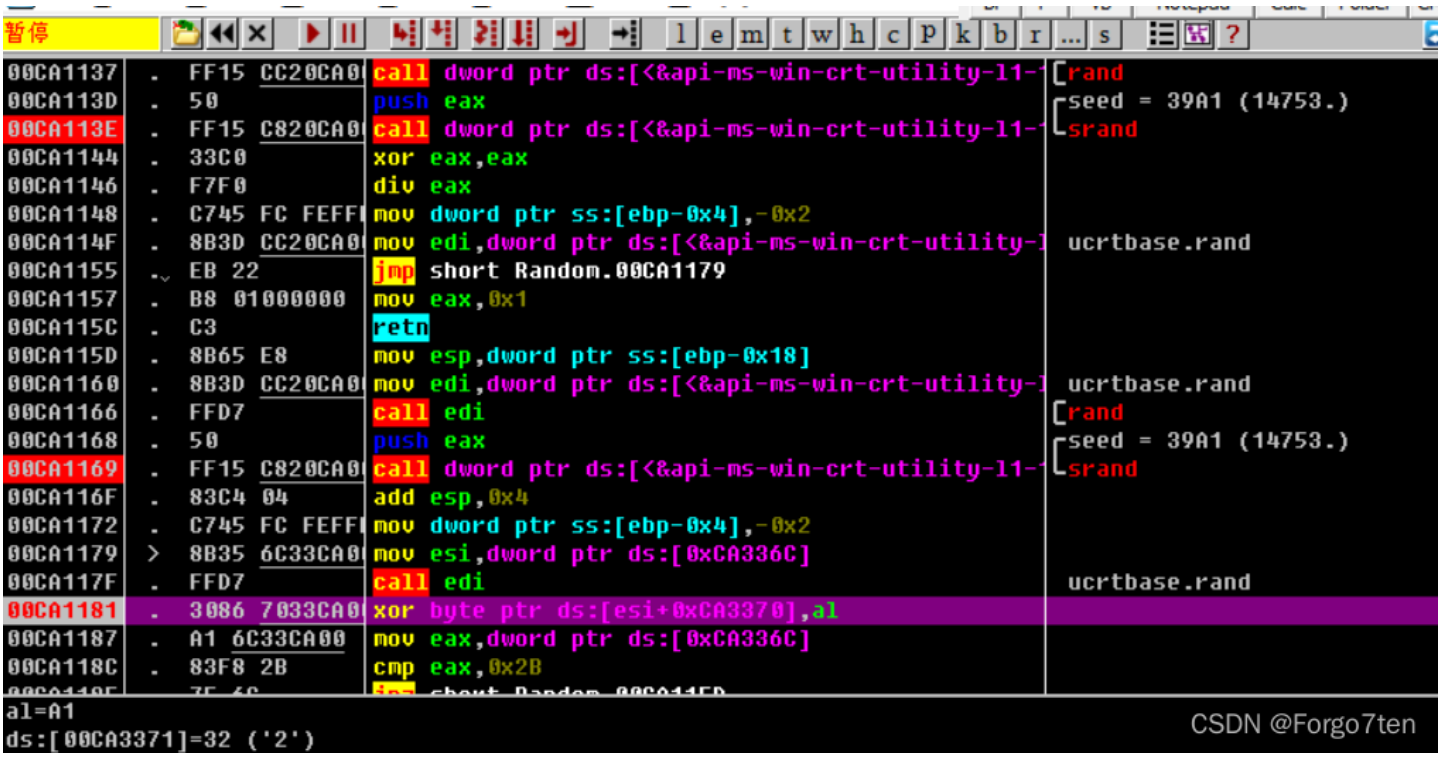
```
.rdata:00CA2518 stru_CA2518     dd 0FFFFFFFEh           ; GSCookieOffset
.rdata:00CA2518                                         ; DATA XREF: _main+5↑o
.rdata:00CA2518                 dd 0                    ; GSCookieXOROffset
.rdata:00CA2518                 dd 0FFFFFFD8h           ; EHCookieOffset
.rdata:00CA2518                 dd 0                    ; EHCookieXOROffset
.rdata:00CA2518                 dd 0FFFFFFFEh           ; ScopeRecord.EnclosingLevel
.rdata:00CA2518                 dd offset loc_CA1157    ; ScopeRecord.FilterFunc
.rdata:00CA2518                 dd offset loc_CA115D    ; ScopeRecord.HandlerFunc
.rdata:00CA2534                 align 8
.rdata:00CA2538 stru_CA2538     dd 0FFFFFFFEh           ; GSCookieOffset
.rdata:00CA2538                                         ; DATA XREF: __scrt_common_main_seh(void)+2↑o
.rdata:00CA2538                 dd 0                    ; GSCookieXOROffset
.rdata:00CA2538                 dd 0FFFFFFCCh           ; EHCookieOffset
.rdata:00CA2538                 dd 0                    ; EHCookieXOROffset
.rdata:00CA2538                 dd 0FFFFFFFEh           ; ScopeRecord.EnclosingLevel
.rdata:00CA2538                 dd offset loc_CA1445    ; ScopeRecord.FilterFunc
.rdata:00CA2538                 dd offset loc_CA1459    ; ScopeRecord.HandlerFunc
.rdata:00CA2554                 align 8
.rdata:00CA2558 stru_CA2558     dd 0FFFFFFFEh           ; GSCookieOffset
.rdata:00CA2558                                         ; DATA XREF: ___scrt_is_nonwritable_in_current_image+2↑o
.rdata:00CA2558                 dd 0                    ; GSCookieXOROffset
.rdata:00CA2558                 dd 0FFFFFFD8h           ; EHCookieOffset
.rdata:00CA2558                 dd 0                    ; EHCookieXOROffset
.rdata:00CA2558                 dd 0FFFFFFFEh           ; ScopeRecord.EnclosingLevel
.rdata:00CA2558                 dd offset loc_CA1767    ; ScopeRecord.FilterFunc
.rdata:00CA2558                 dd offset loc_CA177A    ; ScopeRecord.HandlerFunc
.rdata:00CA2574 __IMPORT_DESCRIPTOR_VCRUNTIME140 dd rva off_CA265C ; Import Name Table
.rdata:00CA2578                 dd 0                    ; Time stamp
.rdata:00CA257C                 dd 0                    ; Forwarder Chain
.rdata:00CA2580                 dd rva aVcruntime140Dl  ; DLL Name
00001918 00CA2518: .rdata:stru_CA2518 (Synchronized with Hex View-1)
```


可以断定出现div 0异常后会执行loc_CA115D的逻辑

main函数是一个递归，根据dword_CA336C==43来结束递归，且不等于0时不输入字符

```
13  const char **v13; // [esp+0h] [ebp-28h]
14  const char **v14; // [esp+4h] [ebp-24h]
15
16  if ( !dword_CA336C )
17    sub_CA10B0(v3, "%s", (char)user_input);
18  v4 = rand();
19  srand(v4);
20  v5 = dword_CA336C;
21  user_input[v5] ^= rand();
22  if ( dword_CA336C == 43 )
23  {
24    dword_CA336C = 0;
25    v6 = 0;
26    v7 = 0;
27    v8 = 0;
28    v9 = 0;
29    do
30    {
31      if ( user_input[v9] != data[v8] )
32        break;
33      v8 = ++v6;
34      v7 = v6;
35      dword_CA336C = v6;
36      v9 = v6;
37    }
38    while ( v6 < 42 );
39    v10 = "fake input..\n";
40    if ( v7 == 42 )
41      v10 = "congratulation!\n";
42    sub_CA1050(v7, v10, v12);
43    result = 0;
44  }
45  else
46  {
47    ++dword_CA336C;
48    main(v12, v13, v14);
49    result = 0;
50  }
51  return result;
52}
```

主功能就是将输入的字符和随机数进行异或，重点是找准srand的时机



三处下断点，发现main函数的流程就是执行两次srand(rand())后，再和rand()进行异或

exp为

```c
#include <stdio.h>
#include "mycrypto.h"
#include <ctype.h>

int main() {
    char data[44] = {
    0x3E, 0xCD, 0xAA, 0x8E, 0x96, 0x1F, 0x89, 0xCD, 0xDB, 0xF1, 0x70, 0xF2, 0xA9, 0x9C, 0xC2, 0x8B,
    0xF2, 0xFE, 0xAD, 0x8B, 0x58, 0x7C, 0x2F, 0x03, 0x4A, 0x65, 0x31, 0x89, 0x76, 0x57, 0x88, 0xDF,
    0xB8, 0xE9, 0x01, 0xE9, 0xDE, 0xE5, 0x86, 0x68, 0x8F, 0x24, 0xD3, 0x5A
    };
    int nxor = 0;
    int s1 = 0;
    int s2 = 0;

    for (int i = 0; i < 44; ++i) {
        s1 = rand();
        printf("srand-1 =%02x\n", s1);
        srand(s1);

        s2 = rand();
        printf("srand-2 =%02x\n",i,s2);
        srand(s2);

        nxor = rand();
        printf("xor =%02x\n", nxor);
        data[i] ^= nxor;

    }
    printf("\n\n");

    puts(data);

}
```

运行得到

```
srand-1 =29
srand-2 =00
xor =258
srand-1 =53bd
srand-2 =01
xor =39a1
srand-1 =22e6
srand-2 =02
xor =74cb
srand-1 =4613
srand-2 =03
xor =49e9
srand-1 =6145
srand-2 =04
xor =49ed
srand-1 =641b
srand-2 =05
xor =682c
srand-1 =40d3
srand-2 =06
xor =11ec
srand-1 =ce1
srand-2 =07
xor =9fb
```

```
xor =91b
srand-1 =387a
srand-2 =08
xor =38e9
srand-1 =1ad5
srand-2 =09
xor =1ec4
srand-1 =34be
srand-2 =0a
xor =1116
srand-1 =12a8
srand-2 =0b
xor =4797
srand-1 =60ad
srand-2 =0c
xor =4399
srand-1 =b12
srand-2 =0d
xor =76b1
srand-1 =3db4
srand-2 =0e
xor =70a4
srand-1 =2781
srand-2 =0f
xor =3e9
srand-1 =03
srand-2 =10
xor =c3
srand-1 =7d7b
srand-2 =11
xor =54c6
srand-1 =2507
srand-2 =12
xor =b80
srand-1 =3516
srand-2 =13
xor =14bf
srand-1 =24a1
srand-2 =14
xor =73e
srand-1 =6dc6
srand-2 =15
xor =4f44
srand-1 =435f
srand-2 =16
xor =2d18
srand-1 =62fb
srand-2 =17
xor =5c2e
srand-1 =1605
srand-2 =18
xor =6b73
srand-1 =6737
srand-2 =19
xor =956
srand-1 =2c53
srand-2 =1a
xor =3752
srand-1 =7371
srand-2 =1b
```

```
xor =bb8
srand-1 =7ed9
srand-2 =1c
xor =635b
srand-1 =1afc
srand-2 =1d
xor =2066
srand-1 =32f2
srand-2 =1e
xor =7ded
srand-1 =3bbe
srand-2 =1f
xor =5bbc
srand-1 =5883
srand-2 =20
xor =6c8a
srand-1 =30e0
srand-2 =21
xor =67d8
srand-1 =560b
srand-2 =22
xor =5236
srand-1 =2dd9
srand-2 =23
xor =478f
srand-1 =3b92
srand-2 =24
xor =59e6
srand-1 =3a88
srand-2 =25
xor =4ed3
srand-1 =5eb7
srand-2 =26
xor =2eb1
srand-1 =3c14
srand-2 =27
xor =5f51
srand-1 =6e91
srand-2 =28
xor =57b9
srand-1 =11
srand-2 =29
xor =159
srand-1 =8d5
srand-2 =2a
xor =5ed3
srand-1 =61f7
srand-2 =2b
xor =515a


flag{3e625fe0-fb18-4f87-93c1-1ec217f86796}
```