获取SSDT地址（Win10 1803）

原创

月初网恋月底分 于 2020-07-27 23:37:19 发布 511 收藏 2

一、特征码搜索SSDT（Win10 1803）

- 读取 C0000082 寄存器`rdmsr c0000082`，得到 KiSystemCall64Shadow函数地址：

```
0: kd> rdmsr c0000082
msr[c0000082] = fffff807`0b54f140
```

```
nt!KiSystemCall64Shadow:
fffff807`0b54f140 0f01f8          swapgs
fffff807`0b54f143 654889242510700000 mov   qword ptr gs:[7010h],rsp
fffff807`0b54f14c 65488b242500700000 mov   rsp,qword ptr gs:[7000h]
fffff807`0b54f155 650fba24251870000001 bt  dword ptr gs:[7018h],1
fffff807`0b54f15f 7203            jb      nt!KiSystemCall64Shadow+0x24 (fffff807`0b54f164)  Branch
```

- 从KiSystemCall64Shadow的地址向上查找（不超过0x200000的范围内），可以直接找到KeServiceDescriptorTable和KeServiceDescriptorTableShadow的地址。
- 特征码是【4C 8D 15 XX XX XX XX 4C 8D 1D XX XX XX XX】

```
nt!KiSystemServiceRepeat:
fffff807`0b3e1104 4c8d1575a73100  lea     r10,[nt!KeServiceDescriptorTable (fffff807`0b6fb880)]
fffff807`0b3e110b 4c8d1d6e383000  lea     r11,[nt!KeServiceDescriptorTableShadow (fffff807`0b6e4980)]
fffff807`0b3e1112 f7437880000000  test    dword ptr [rbx+78h],80h
fffff807`0b3e1119 7413            je      nt!KiSystemServiceRepeat+0x2a (fffff807`0b3e112e)  Branch
```

二、代码实现

```
#include "SSDT.h"
#include <intrin.h>
#include <wdf.h>
#include <ntddk.h>
//***********************************
// Method:      SSDT::SSDTfind
// Description: 获取SSDT的地址
// Returns:     PVOID -
//***********************************
PVOID SSDT::SSDTfind()
{
 PUCHAR pKiSystemCall64Shadow = (PUCHAR)__readmsr(0xc0000082);  //rdmsr c0000082 , 定位KiSystemCall64Shadow函数（Win10 1803）
 PUCHAR startAddr = pKiSystemCall64Shadow - 0x200000;  //向上查找0x200000范围
 PUCHAR endAddr = pKiSystemCall64Shadow;

 PUCHAR pSSDTpattern = 0;
 int count = 0; //成功匹配特征码的地址个数
 UCHAR b0 = 0, b1 = 0, b2 = 0, b7 = 0, b8 = 0, b9 = 0;
 for (PUCHAR i = startAddr; i < endAddr; i++)
 {
  b0 = *i;
  b1 = *(i + 1);
  b2 = *(i + 2);
  b7 = *(i + 7);
  b8 = *(i + 8);
  b9 = *(i + 9);
  // 特征码【4C 8D 15 XX XX XX XX 4C 8D 1D XX XX XX XX】
  if (b0 == 0x4C && b1 == 0x8d && b2 == 0x15 && b7 == 0x4C && b8 == 0x8d && b9 == 0x1D)
  {
   DbgPrint("Find SSDT pattern On = %p \n", i);
   count++;
   pSSDTpattern = i;
  }
 }

 //如果特征码匹配失败或者匹配多个，则查找SSDT失败
 if (count = 0 || count > 1) { return 0; }

 //要跳转的函数地址= 下一条指令的地址 +  Operand
 PVOID pSSDT = (pSSDTpattern + 7) + *(signed int*)(pSSDTpattern + 3);
 return pSSDT;
}
```

参考资料

- [分享]分享一个获取KeServiceDescriptorTable的方法