# 来自看雪的手把手调试DebugPort清零

转载

H-KING 于 2017-06-30 10:26:42 发布 
分类专栏： 驱动编程


驱动编程 专栏收录该内容
43 篇文章 4 订阅
订阅专栏

现在多数程序为了防止调试。基本上都用到了驱动HOOK 内核API。
至于绕过那些HOOK，基本上大家应该已经是没有什么问题了。

估计像我这样的菜鸟也不算多了。研究DebugPort 清0，倒是难倒了我。。。

所谓DebugPort 清0，就是向 EPROCESS->DebugPort 不停写入 NULL（0）值。。让调试器无法收到调试信息。。。

现在能找到的资料也不算多。。。也许是因为解决方法一旦公布出来，从而导致程序升级。

就现在能找到的资料来谈谈;

1：通过修改调试相关函数，修改DebugPort端口，指向EPROCESS结构中的其它成员，大多数驱动已经有监视了。

2：NOP掉程序驱动中清0函数和监视函数，需要分析程序驱动。

3：修改EThread->Process 指向自定义结构，程序驱动如果也是这样访问的DebugPort此方法都无效了。

4：HOOK 缺页异常然后把eprocess弄成invalid然后自己处理，此方法本菜鸟没试过，因为本菜鸟还不知如果HOOK 缺页异常，能找到的资料甚少。也希望有大牛们能指点一二。

5：用WRK自己编译内核ntoskrnl.exe，改造EPROCESS结构。。。此法只能用于 XP64和2003SP1.

暂时本菜只了解到这些方法，本菜才疏学浅，至今自己也没想出别的解决思路。。。

下面说下本菜用第一种方法在 WIN7下的研究。。

因为网上资料大多数为XP系统的。所以自己分析了一下WIN7 调试相关函数。

本菜用的双机调试，至于如何双机调试网上资料很多。就不在赘诉。

打开虚拟机，打开一个程序。如 LoadSys.exe

之后本机打开Windbg通过串行端口连接虚拟机。

lkd->!process 0 0 LoadSys.exe

得到LoadSys.exe 的EPROCESS地址如。0x87654321

lkd->ba r4 0x87654321+0xec （WIN7上DebugPort 偏移为 0xec,可以通过lkd->dt nt!_eprocess,查看。）

上一句下了访问断点。。之后进入虚拟机，打开一个OD。附加LoadSys.exe..这时虚拟机就会断下来。我们来看看调用了哪些函数。。

代码:

```
读： nt!DbgkCreateThread+0x22a: 856ab86a 399eec000000    cmp     dword ptr [esi+0ECh],ebx
nt!PsGetProcessDebugPort+0x8: 85514130 8b80ec000000    mov     eax,dword ptr [eax+0ECh]
nt!DbgkpSetProcessDebugObject+0x8d: 856f7959 83beec00000000  cmp     dword ptr [esi+0ECh],0
nt!DbgkpSetProcessDebugObject+0x9d: 856f7969 89beec000000    mov     dword ptr [esi+0ECh],edi
nt!DbgkpMarkProcessPeb+0x85: 856f6e90 3987ec000000    cmp     dword ptr [edi+0ECh],eax
nt!DbgkpQueueMessage+0xad: 856f74a7 8b80ec000000    mov     eax,dword ptr [eax+0ECh]
nt!KiDispatchException+0x1d8: 8550539f 39b0ec000000    cmp     dword ptr [eax+0ECh],esi
nt!DbgkForwardException+0x49: 8565deac 8b98ec000000    mov     ebx,dword ptr [eax+0ECh]
nt!PspExitThread+0x2ad: 8569094c 83bfec00000000  cmp     dword ptr [edi+0ECh],0 nt!DbgkExitThread+0x28:
856f8cf1 83b9ec00000000  cmp     dword ptr [ecx+0ECh],0 nt!PspTerminateAllThreads+0x1dd: 856a7ff4
83bfec00000000  cmp     dword ptr [edi+0ECh],0 nt!DbgkExitProcess+0x28: 856f8d63 83b9ec00000000  cmp
 dword ptr [ecx+0ECh],0 nt!DbgkpCloseObject+0xd6: 856f707f 3998ec000000    cmp     dword ptr
[eax+0ECh],ebx nt!DbgkpCloseObject+0x119: 856f70c2 3998ec000000    cmp     dword ptr [eax+0ECh],ebx
nt!DbgkpCloseObject+0x121: 856f70ca 83a0ec00000000  and     dword ptr [eax+0ECh],0
```

以上为读断点部分。。

记下函数+偏移,进入虚拟机，退出OD和LoadSys.exe...

重新打开LoadSys.exe
之后回到Windbg
lkd->!process 0 0 LoadSys.exe
重新获得 LoadSys.exe 的EPROCESS地址，如：0x88776655

lkd->ba w4 0x88776655+0xec
给LoadSys.exe的DebugPort下写入断点

再进入虚拟机，开OD，附加LoadSys.exe..之后就会断下。。。

代码:

```
写： nt!DbgkpSetProcessDebugObject+0xa3   //和读其中一个重复 nt!DbgkClearProcessDebugObject+0x41: 856d3e84
05ec000000      add     eax,0ECh nt!DbgkpCloseObject+0x128            和读其中一个重复
```

至此我们就找到了WIN7下 调试写入DebugPort的内核相关函数。

剩下的事情，我们只需要编写驱动将每个函数偏移下的 ec 改写成 其它EPROCESS中成员 偏移。。。如WIN7下的 CREATE_TIME成员 为 a0。。。EXIT_TIME成员为a8...

改写完后。。。用OD随便附加一个程序。查看其EPROCESS中 你改写的那个偏移，如我改写的是a0。。a0中就存有一个 ？？ 对象句柄？？。。就证明改写成功。。。

然后打开有保护驱动的XX程序。。。用OD附加。OD创建线程完后。加载DLL的时候。。直接蓝屏。。。

代码:

```
a5f3334e 894dfc          mov     dword ptr [ebp-4],ecx a5f33351 eb27            jmp     EagleNT+0xa37a
(a5f3337a) a5f33353 833d5833f4a506  cmp     dword ptr [EagleNT+0x1a358 (a5f43358)],6 a5f3335a 7517
      jne     EagleNT+0xa373 (a5f33373) a5f3335c 833d5c33f4a501  cmp     dword ptr [EagleNT+0x1a35c
(a5f4335c)],1 a5f33363 750e            jne     EagleNT+0xa373 (a5f33373) a5f33365 8b5508          mov
  edx,dword ptr [ebp+8] a5f33368 81c22c020000    add     edx,22Ch a5f3336e 8955fc          mov
dword ptr [ebp-4],edx a5f33371 eb07            jmp     EagleNT+0xa37a (a5f3337a) a5f33373
c745fc00000000  mov     dword ptr [ebp-4],0 a5f3337a 837dfc00        cmp     dword ptr [ebp-4],0
a5f3337e 7415            je      EagleNT+0xa395 (a5f33395) a5f33380 6a10            push    10h
a5f33382 8b45fc          mov     eax,dword ptr [ebp-4] a5f33385 8b08            mov     ecx,dword ptr
[eax]  ds:0023:0000022a=????????    a5f33387 51              push    ecx a5f33388 6820bdf4a5      push
  offset EagleNT+0x22d20 (a5f4bd20) a5f3338d e89ea1ffff      call    EagleNT+0x4530 (a5f2d530) a5f33392
8845fb          mov     byte ptr [ebp-5],al a5f33395 8a45fb          mov     al,byte ptr [ebp-5]
a5f33398 8be5            mov     esp,ebp a5f3339a 5d              pop     ebp a5f3339b c20400
ret     4
```

分析DUMP文件蓝在这句。。。应该是游戏有检测。。。

a5f33385 8b08  mov  ecx,dword ptr [eax]  ds:0023:0000022a=????????

于是HOOK 程序驱动跳过这里。。。

仍然蓝屏，分析DUMP文件蓝在

ntkrnlpa.exe ( nt+2af571 )

代码:

```
856c054a 33ff            xor     edi,edi 856c054c 894b1c          mov     dword ptr [ebx+1Ch],ecx
856c054f 3bc7            cmp     eax,edi 856c0551 750d            jne     nt+0x2af560 (856c0560)
856c0553 c7442414530300c0 mov     dword ptr [esp+14h],0C0000353h 856c055b e9ad000000      jmp
nt+0x2af60d (856c060d) 856c0560 b101            mov     cl,1 856c0562 8d7010          lea     esi,
[eax+10h] 856c0565 ff155c214185    call    dword ptr [nt+0x115c (8541215c)] 856c056b 88442413
mov     byte ptr [esp+13h],al 856c056f 8bc6            mov     eax,esi 856c0571 f00fba3000      lock
btr dword ptr [eax],0   ds:0023:4d5d748c=???????? 856c0576 7205            jb      nt+0x2af57d
(856c057d) 856c0578 e886c7d7ff      call    nt+0x2bd03 (8543cd03) 856c057d 64a124010000    mov
eax,dword ptr fs:[00000124h] 856c0583 8b542414        mov     edx,dword ptr [esp+14h] 856c0587 894604
       mov     dword ptr [esi+4],eax 856c058a 0fb6442413      movzx   eax,byte ptr [esp+13h] 856c058f
89461c          mov     dword ptr [esi+1Ch],eax 856c0592 f6423801        test    byte ptr [edx+38h],1
```

蓝在这里

856c0571 f00fba3000  lock btr dword ptr [eax],0  ds:0023:4d5d748c=????????

由于本菜才疏学浅，于是本菜到此不知道如何再分析下去。。。

希望有大牛们，指点一二。。除上述5种方法以外，还有哪些方法能够绕过DebugPort 清0.

再就是 本菜的分析如何才能继续。。。。

只要思路，方法。。。不要代码结果。。。本菜虽菜，但会以学习为目的提升自身能力，不愿直接拿到那代码而不明其中原理。。。

转自看雪的一位牛人。在这里对那位牛人表示敬佩