驱动层IAT HOOK

原创

MFCJCK 于 2014-05-05 15:46:36 发布 1141 收藏
分类专栏： 零星点点

还是看雪“新人交流投稿”版块的问题引起的（ http://bbs.pediy.com/showthread.php?t=95230 ）。
这次是有人询问sysnap的“学习笔记之钩住驱动程序导入表 ”（ http://bbs.pediy.com/showthread.php?t=62316 ）中，为何要使用将驱动文件映像进内存再查找其导入表结构，而不是直接使用内存中已有的驱动内容来查找导入表。
因为Sysnap的原文里没有对这个做解释，我又没有做过这个，看了这问题一开始也不明白，WINDBG下Local Kernel Debug看ntkrnlpa.exe内存（我这台的CPU是AMD的……），发现应该是导入表结构的地方，其内容完全不对应，再一看它在INIT段，这才恍然大悟。
这个关键就是驱动的导入表结构放在INIT段了，而INIT段在编译链接时是标为“可丢弃的”，在调用完DriverEntry之后，整个区段就不应该再被访问和使用了。
导入表_IMAGE_IMPORT_DESCRIPTOR结构等，只是为了系统加载驱动时能够正确填充IAT表而存在，加载完毕之后，只需要IAT表存在，驱动就可以正常使用其中的内核函数地址，除IAT以外的结构内容就是可以丢弃的，因此编译时被放在INIT段。
所以在驱动程序DriverEntry调用完毕后，再对其进行IAT HOOK时，原来的内存中的导入表已经被清除，就必须重新将驱动文件进行映像后才能读取到原来的导入表结构从而得到IAT表相应信息，再根据驱动实际基址进行重定位修正。
但如果是使用PsSetLoadImageNotifyRoutine在驱动文件映像加载过程中进行IAT HOOK，此时DriverEntry还没有被调用，INIT段及其中的导入表结构仍然有效，应该就可以直接使用。