Vista_WIN7驱动完整性校验解析PE蓝屏BUGMicrosoft认证考试 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/590/2021_2022_Vista_WIN7_c100_590069.htm 在VISTA以后的操作系统，系统使用MmLoadSystemImage加载驱动之前，会调用MmCheckSystemImage函数来检查镜像正确性，在VISTA及以后的操作系统中，MmCheckSystemImage发生了一个有意思的变化. 原本MmCheckSystemImage(vista以前的系统上)，会使用SEC_IMAGE作为 AllocationAttributes来调用ZwCreateSection为驱动文件创建Section，但是VISTA以后的系统上，该参数被换成了一个未公开的值: 0x100000(注意,SEC_IMAGE是0x1000000，6个0)。 在ZwCreateSection 中，系统会检查如果调用线程的上个模式不是KernelMode,则不允许使用这个未公开的AllocationAttributes: NtCreateSection: .....无关部分..... //取当前线程上个模式 if (KeGetPreviousMode() != KernelMode) { //如果是用户模式，如果Attributes有0x2000000或0x10000的话，则返回无效参数 v13 = 0. if (!(Attributes gt.MmCreateSection(这个函数在VISTA开始发生了巨大的变化)中，进行一些判断后(包括这个特殊的Attributes)，系统会调用MiValidateImageHeader进行镜像检查，此时系统会将这个镜像通过 MiMapImageInSystemCache函数map到系统缓存中，然后将按镜像的页数 * PAGE_SIZE大小的，分配分页内存，并将PE数据COPY到内存中 接着系统会调用SeValidateImageHeader函数，这个函数只是简单地为_g_CiCallbacks中存放的函数准备函数，便调用_g_CiCallbacks存放的函数。_g_CiCallbacks这个全局变量中

存放着系统初始化时 (SeInitSystem-gt.SepInitializeCodeIntegrity) 存入的 ci.dll的CiValidateImageHeader函数。
CiValidateImageHeader首先会对镜像进行一些检查工作，然后开始调用CipValidateFileHash函数，CipValidateFileHash函数对文件做一些解析工作后，开始调用CipImageGetImageHash，此函数会分析PE的每一个节，并对其节内数据调用SHA签名算法函数A_SHAUpdate。 注意前面加粗的文字，由于是按整页数来分配和COPY数据的，因此如果某一个节的数据长度(SizeOfRawData)超出了页对齐的范围(MiMapImageInSystemCache似乎并不将这个数据算成一个新的节)，那么A_SHAUpdate中的数据COPY函数将触及到未分配内存，从而引发BSOD。 这里提供一个简单的例子(下载到：http://www.debugman.com/read.php?tid=3217帖子中的附件或http://mj0011.ys168.com 漏洞演示目录下bsodxx.rar) 这个PE文件的最后一个节的SizeOfRawData是0x1004，使用任意一个加载工具加载此文件，系统将立即BSOD，BSOD时的Stack类似： kd&gt. kc nt!MmAccessFault nt!KiTrap0E nt!memcpy CI!A_SHAUpdate CI!CipImageGetImageHash CI!CipValidateFileHash CI!CiValidateImageHeader nt!SeValidateImageHeader nt!MiValidateImageHeader nt!MmCreateSection nt!NtCreateSection nt!KiFastCallEntry nt!ZwCreateSection nt!MmCheckSystemImage nt!MiCreateSectionForDriver nt!MiObtainSectionForDriver nt!MmLoadSystemImage nt!IopLoadDriver nt!IopLoadUnloadDriver 因此说，解析PE很危险啊很危险。 更多优质资料尽在百考试题论坛 百考试题在线题库 微软认证更