深入浅出分析Linux系统内核漏洞的问题（2） PDF转换可能丢失图片或格式，建议阅读原文
https://www.100test.com/kao_ti2020/144/2021_2022__E6_B7_B1_E5_85_A5_E6_B5_85_E5_c103_144392.htm 拒绝服务类漏洞 拒绝服务攻击是目前比较流行的攻击方式，它并不取得服务器权限，而是使服务器崩溃或失去响应。对Linux的拒绝服务大多数都无须登录即可对系统发起拒绝服务攻击，使系统或相关的应用程序崩溃或失去响应能力，这种方式属于利用系统本身漏洞或其守护进程缺陷及不正确设置进行攻击。 另外一种情况，攻击者登录到Linux系统后，利用这类漏洞，也可以使系统本身或应用程序崩溃。这种漏洞主要由程序对意外情况的处理失误引起，如写临时文件之前不检查文件是否存在，盲目跟随链接等。 下面，我们简单描述一下Linux在处理intel IA386 CPU中的寄存器时发生错误而产生的拒绝服务漏洞。该漏洞是因为IA386多媒体指令使用的寄存器MXCSR的特性导致的。由于IA386 CPU规定MXCSR寄存器的高16位不能有任何位被置位，否则CPU就会报错导致系统崩溃。为了保证系统正常运转，在linux系统中有一段代码专门对MXCSR的这个特性作处理，而这一段代码在特定的情况下会出现错误，导致MXCSR中的高16位没有被清零，使系统崩溃。如果攻击者制造了这种“ 极限 ”的内存情况就会对系统产生DoS效果。 攻击者通过调用get_fpxregs函数可以读取多媒体寄存器至用户空间，这样用户就可以取得MXCSR寄存器的值。调用set_fpxregs函数可以使用用户空间提供的数据对MXCSR寄存器进行赋值。通过对MXCSR的高16位进行清0，就保证了IA386 CPU的这个特性。如果产生一种极限效果使程序跳过

这一行，使MXCSR寄存器的高16位没有被清0，一旦MXCSR寄存器的高16位有任何位被置位，系统就会立即崩溃！因为利用这个漏洞攻击者还需要登录到系统，这个漏洞也不能使攻击者提升权限，只能达到DoS的效果，所以这个漏洞的危害还是比较小的。但是分析这个漏洞就没有意义了吗？其实由分析这个漏洞可以看出：Linux内核开发成员对这种内存拷贝时出现错误的情况没有进行考虑，以至造成了这个漏洞，分析了解了这个漏洞后，在漏洞挖掘方面也出现了一种新的类型，使我们在以后的开发中可以尽量避免这种情况。 接下来让我们看一种Linux内核算法上出现的漏洞。先来简单介绍一下这个漏洞，当Linux系统接收到攻击者经过特殊构造的包后，会引起hash表产生冲突导致服务器资源被耗尽。这里所说的hash冲突就是指：许多数值经过某种hash算法运算以后得出的值相同，并且这些值都被储存在同一个hash槽内，这就使hash表变成了一个单向链表。而对此hash表的插入操作会从原来的复杂度O(n)变为O(n*n)。这样就会导致系统消耗巨大的cpu资源，从而产生了DoS攻击效果。 我们先看一下在linux中使用的hash算法，这个算法用在对Linux route catch的索引与分片重组的操作中。在今年五月Rice University计算机科学系的Scott A. Crosby与Dan S. Wallach提出了一种新的低带宽的DoS攻击方法，即针对应用程序所使用的hash算法的脆弱性进行攻击。这种方法提出：如果应用程序使用的hash算法存在弱点，也就是说hash算法不能有效地把数据进行散列，攻击者就可以通过构造特殊的值使hash算法产生冲突引起DoS攻击。 