Spectre攻击分析

hades / 2018-01-10 09:34:55 / 浏览数 3027 安全技术 漏洞分析 顶(0) 踩(0)

Author:■■■■■■■■

事件：

Spectre
事件的爆发，不仅仅让芯片厂商忙碌，而更是让安全研究人员无比兴奋，感觉就像是安全领域又被发现了一片处女地，让安全研究人员找到了新的探索领域；下面是我们针对

概述：

现代处理器使用分支预测和投机执行来提高性能。比如一个条件语句的判断取决于内存中的一个值，cpu在内存读取到这个值的几个周期内会保存一个检测点，然后猜测一个

Spectre攻击则是当一个恶意程序利用上面这个投机执行的机制，当我们构造恶意的分支来让投机执行机制去执行，虽然在cpu的会通过上面的检测点来纠正错误，但是恶意

```
if (x < array1_size)
y = array2[array1[x] * 256];
```

`arrary1_size`表示`arrary1`数组（定义为`uint8_t`）的大小，如果我们在多次让x不会越过数组1的大小来让提升cpu投机执行会选择下面分支的概率，这是我们想窃取内
*
`256]`会被放到缓存里，如果我们用flush+reload的技术遍历`arrary2[i*512]`，来确定其中那个i所对于的内存地址被放入缓存里，i的值就是地址`0x1234`的值。这样我们完

Cpu Cache

Cpu Cache是解决处理的时钟周期远大于内存时钟周期的方案。这里只做简单介绍具体内容可参考[2],[3],[4]。Cache是以Cach
line存储内容,intel目前每个处理器中Cache line为64字节，Cache 采用16-Way Set Associative方式关联内存地址.将内存地址的低6位位cache
line中的偏移，中间12位 set的偏移,详见[2]。其余高地址为tag。

侧信道攻击

本文只分析[1]中的flush+reload的方法，具体介绍参考[5]，大体思路是做一个上面代码中的arrary2这样的大数组，然后测量arrary2数组中的那个偏移的地址被放入缓存中

POC分析：

具体代码在[1]中的附录内。这里只讲一下大体利用思路

首先如图1所示：

图1

100行首先计算我们要窃取内容的地址与arrary1数组的相对偏移。Len是内容的长度。攻击代码是readMemoryByte函数，参数1就是相对偏移，value是窃取的内容，score

第二部分flush+reload技术需要的数组和cpu 投机执行的代码

其中unused1和unused2是64字节防止arrary1和araary2还有arrary_size落入一个cacheline。512是尽量不要让后面的攻击代码落入一个set里。读者可以参考[2]里面的内

第三部分 flush+reload技术

首先是定义CACHE_HIT_THRESHOLD 此值是检测内存地址没有放入缓存时程序的执行时间和有被放入缓存时执行时间的差的参考。

其中

`_mm_flush`时`cflush`指令的c库函数。59行到61的算法时多次让提升`victim_function`分支正确的的概率，当`j%6`为`0`时，行61执行后x就是`malicious_x`，`maliciou

下面是获取偏移的代码

大体思路就是遍历arrary2中内存中的值，只不过步长是`i*512`，测量取值的时间。来判断那个偏移被放入缓存中，其中行69换成`mix_i=i`就好理解了，我测试了一些没有行

最后一步统计偏移的准确率

参考

[1]https://spectreattack.com/spectre.pdf

[2]https://coolshell.cn/articles/10249.html

[3] http://blog.jobbole.com/85185/

[4][https://software.intel.com/sites/default/files/managed/a4/60/325384-sdm-vol-3abcd.pdf 第11章

[5] FLUSH+RELOAD: a high resolution, low noise, L3 cache side-channel attack

上一篇：Apache Batik XXE—... 下一篇：DeDecms(织梦CMS)最新版...

1. 0 条回复
   - 动动手指，沙发就是你的了！