1. 研究背景

面向返回编程（Return-Oriented Programming，简称ROP ）是一种代码复用攻击技术。该攻击利用程序中原有的程序片段来达到攻击目的。ROP攻击通常分为两个阶段来完成：1）攻击者扫描已有的动态链接库和可执行文件，提取出可以利用以返回指令（后来扩展到其它间接分支指令）结尾的指令片段（也称为配件（gadget））；2）攻击者利用内存漏洞劫持程序控制流，将一系列这样的指令片段串联起来，达到攻击目的。

控制流完整性保护（Control Flow Integrity，简称CFI ）是一种通过限制控制流转移的目标来检测控制流劫持攻击的有效方法。包括ROP攻击在内，所有控制流劫持攻击都需要将程序的控制流重定向到非正常执行预期的位置以达到攻击目的。CFI方法通过运行时监控程序的执行、并将其状态与预先设定好的控制流图（Control-Flow Graph，简称CFG）的路径对比来将程序的控制流限制到有效的执行轨迹中。如果检测到一种无效状态，就会发出警告，通常是终止程序的执行。控制流图通常通过源码分析、二进制分析或者执行剖析得到。

本文提出了一种基于特征的的CFI检测方法，需要利用如下硬件资源：

分支处理单元（Branch Processing Unit，简称BPU）：是现代处理器进行分支指令处理的单元。它主要用于记录分支历史信息，并根据历史信息对后续分支指令的目标地址及分支方向进行预测；

性能监控单元（Performance Monitor Unit，简称PMU ）：是现代处理器引入的用于记录处理器事件的功能单元。现代CPU支持大量的性能监控事件，包含了处理器在运行过程中可能遇到的所有情形，例如指令计数、浮点运算指令计数、L2缓存未命中的时钟周期、分支预测失败事件等。用户可以根据需要设置事件选择寄存器选取需要采样的事件类型。在事件计数过程中，计数器溢出可以出发性能监控中断（Performance Monitor Interrupt，简称PMI），用户可以在中断处理过程中进行相关事件处理；

最近分支记录（Last Branch Record，简称LBR）是X86架构提供的一组用于记录和追踪程序最近的若干次跳转信息的循环寄存器组，它们会记录分支指令的源地址（FROM\_IP）和目标地址(TO\_IP)。这些寄存器的数量与处理器的微架构相关。另外，LBR寄存器具备过滤功能，能够选择性的滤掉一些不需采样的跳转指令。