处理器指令集架构安全中的隐藏指令检测技术项目方案设计

1.项目简介

处理器中指令集类漏洞包括后门和异常指令两种，后门漏洞是指制造商在处理器中遗留的具有高权限的隐藏指令，其可以在用户不知情的情况下执行权限提升、访问机密信息、建立远程连接等敏感操作。异常指令漏洞是指处理器指令在设计逻辑或实现过程中存在缺陷，致使其在执行过程中不能按照既定的设计针对输入提供准确的输出。

在此基础上，为提升处理器芯片底层指令集安全，提出了隐藏指令检测技术，该技术通过开发集成不同指令搜索算法来适配精简和复杂指令集的指令遍历，之后采用单指令执行的方式检测遍历结果并对执行环境进行保护和恢复，从而以较快速度检测不同指令集架构中存在的隐藏指令。

2.项目背景与意义

当前，处理器被广泛的运用于各种领域，同时由于设计集成度的不断提高和规模的不断增大，导致处理器安全问题越来越严重。我国正处于高速发展的阶段，对处理器需求量与日俱增，然而由于外国打压和自身技术能力的限制，我国的处理器严重依赖进口，在安全方面也无法做到自主可控，因此十分有必要对处理器安全漏洞的相关检测方法进行研究。目前，针对于处理器安全漏洞分析与检测技术的研究还处于初级阶段，因此把握时间窗口深入探索这一领域对于处理器行业和国家发展都有着深远的意义。

2.1 近年来 Intel、AMD 等厂商的处理器不断曝出重大安全问题，导致个人和企业近的数据处于风险之中

随着信息技术的发展，计算机、互联网已成为人们日常生产生活中必不可少的部分。近年来，互联网技术不断进步、软件应用不断更新，不断进

化升级的现代处理器给人们带来了许多便捷，但是也带来了不少的安全隐患。1994 年出现在 Pentium 处理器上的 FDIV bug，其会导致浮点数除法出现错误；1997 年 Pentium 处理器被发现存在漏洞，F00F 异常指令可导致 CPU 宕机；2011 年 Intel 处理器可信执行技术(Trusted Execution Technology)存在缓冲区溢出问题，可被攻击者用于权限提升；2017 年 Intel 管理引擎组件中的漏洞可导致远程非授权的任意代码执行；2018 年，Meltdown 和 Spectre 两个 CPU 漏洞几乎影响到过去 20 年制造的每一种计算设备，使得存储在数十亿设备上的隐私信息存在被泄露的风险。这些安全问题严重危害国家网络安全、关键基础设施安全及重要行业的信息安全，已经或者将要造成巨大损失。

2.2 我国芯片产业受制于人，安全性尚未可知，迫切需要处理器安全检测技术

现如今信息安全已经成为互联网关注的热点问题，如何维护互联网安全、如何防止使用计算机信息系统的用户受到不法侵害、如何更有效的维护信息安全等已经成为人们关注的焦点。中央处理器作为计算机系统的运算和控制核心，是信息处理、程序运行的最终执行单元，其安全问题对国家网络、关键基础设施及重要行业的信息安全有着深刻的影响。我国每年需要进口价值 2000 多亿美元的芯片，虽然国产处理器已开始部分应用在关键行业的信息设备中，但近年来美国无端挑起针对中国的贸易战，持续加大对我芯片产业的恶意打压，无论是芯片的设计、原料还是最终的制造环节都收到了较大影响，因此在未来几年内，我国处理器依赖进口的局面不会有根本性改变，短时间内国产处理器仍无法完全替代进口处理器。在国家不断强调信息产业要“自主可控”的背景下，对国内外处理器进行漏洞检测与安全分析的需求十分急迫。

2.3 目前处理器漏洞检测处于起步阶段，核心技术被外国垄断，发展该领域可为国家芯片安全提供重要支撑

由于处理器内部机理并不开源，研究难度大等缘故，当前处理器安全的研究热点主要集中于单个处理器中已知漏洞的发现与防护。现在虽有部分处理器漏洞挖掘的方法及工具，但是这些工具大部分只支持某类甚至是某个处理器漏洞的检测，覆盖不完整，同时在处理器指令集测试的正确率和效率上有一定的缺陷，实用性不强。因此研究多架构处理器漏洞挖掘技术不但可以填补我国在这一方面的空白，还可以为国产处理器提供理论和实际指导，使得处理器中的漏洞被早发现，早分析，早修复，完善国产处理器的安全机制，提高安全标准，从而提升其在国内和国际上的竞争力，助力国产芯片早日打破外国垄断。从长远来看，发展研究多架构处理器漏洞挖掘技术，可以更有效的掌握国内外芯片的安全问题，对我国未来国防安全有着重要的积极意义。

3.项目研究路线

研究内容整体架构如下图所示，基于隐藏指令检测的执行流程，首先面向目前主流的处理器指令集架构研究其架构的基础理论，然后进一步研究 x86、ARM、SPARC、RISC-V 等多种架构下基于模糊测试的隐藏指令检测方式以及高效的指令搜索算法。对于检测出的隐藏指令，需要进行分类去冗余与真实性验证的工作。在上述研究的基础上继续进行隐藏指令的功能分析和后门检测。最后，根据检测出的隐藏指令，对可疑程序进行二进制分析，从而实现隐藏指令的威胁检测。

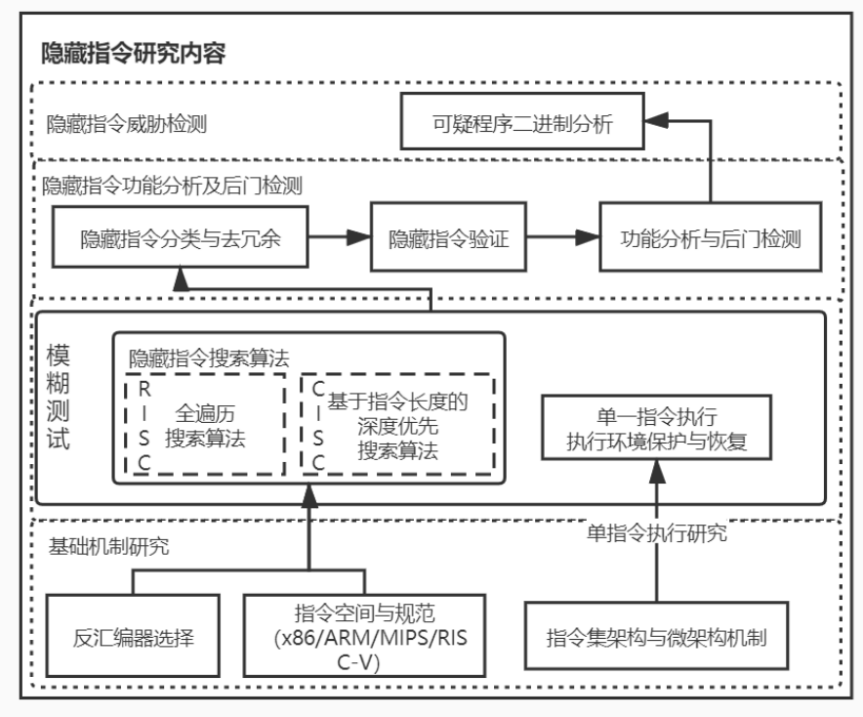

图 1：研究内容总体架构图

3.1 多指令集架构下基于单指令模糊测试的隐藏指令检测技术

在模糊测试单指令执行前后，其执行环境往往存在改变，有时导致系统崩溃或写入测试程序内存等问题。在执行过程中，由于变长指令集的指令集搜索域过大，难以在短时间内完成对指令集的搜索。针对上述的问题，提出了多指令集架构下基于单指令模糊测试的隐藏指令检测技术。

隐藏指令检测方法流程图如下图所示，首先根据被测试处理器平台信号等信息初始化测试系统，搜索待测试指令，使用反汇编器将被测试指令反汇编，同时将指令放在内存中进行单指令运行。对比反汇编与单指令运行结果，如果反汇编器不识别，但是能够成功运行的指令判定为隐藏指令，将其记录在文件中用于进一步分析。测试完成一条指令后，继续搜索下一条带测试指令进行测试。

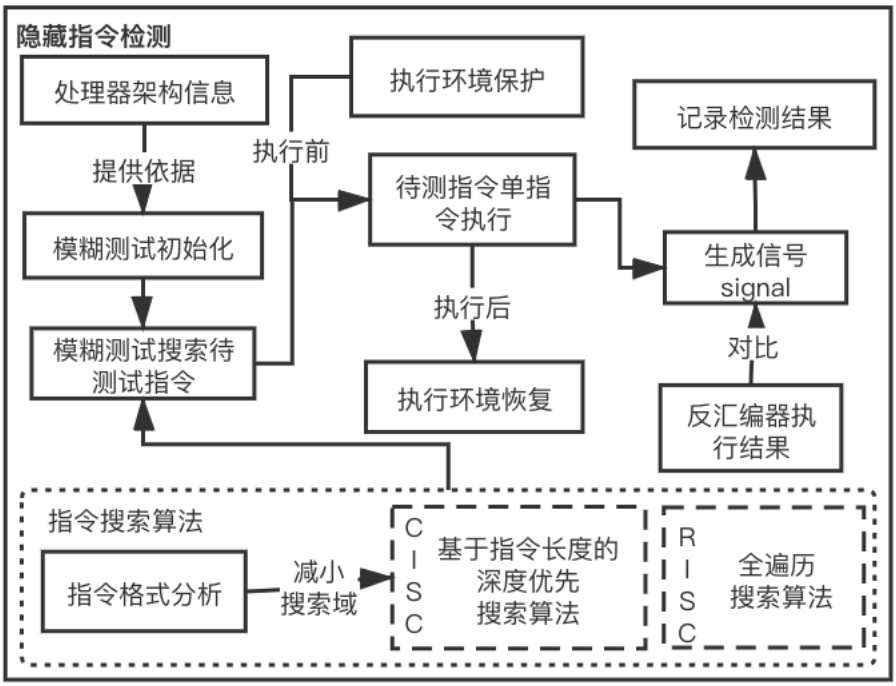

图 2 隐藏指令模糊测试框架图

整个测试过程基于模糊测试的方式，通过指令搜索算法将隐藏指令出现可能性较大的指令或指令区间进行遍历，放在内存中的特定区域。在执行之前首先保存执行前的运行环境，设置单步执行寄存器使得指令在执行后可以把控制权重新移交回测试程序，再进一步恢复之前保存的运行环境。通过对比反汇编器与执行结果判断被测试指令是否为隐藏指令。

隐藏指令检测研究主要包括两个方面：模糊测试的单指令执行机制和基于各架构指令集特性的指令搜索算法。

3.2 基于执行环境监控与执行端口统计的隐藏指令功能分析与后门检测技术

隐藏指令功能分析与后门检测是在隐藏指令被检测后的基础上，在各处理器指令集架构下对隐藏指令进一步的分析研究。在检测出的隐藏指令基础上，首先对基于特定处理器指令集格式下的隐藏指令去冗余与分类算法研究。分类完成后，对指令集的隐藏指令的真实性研究。最后，在各处理器指令集架构下对于真实的隐藏指令建立功能分析模型和进一步的后门检测模型研究。从而达

到对多架构处理器进行指令检测、分类、验证、功能分析及后门检测的效果。整体架构如下图所示：

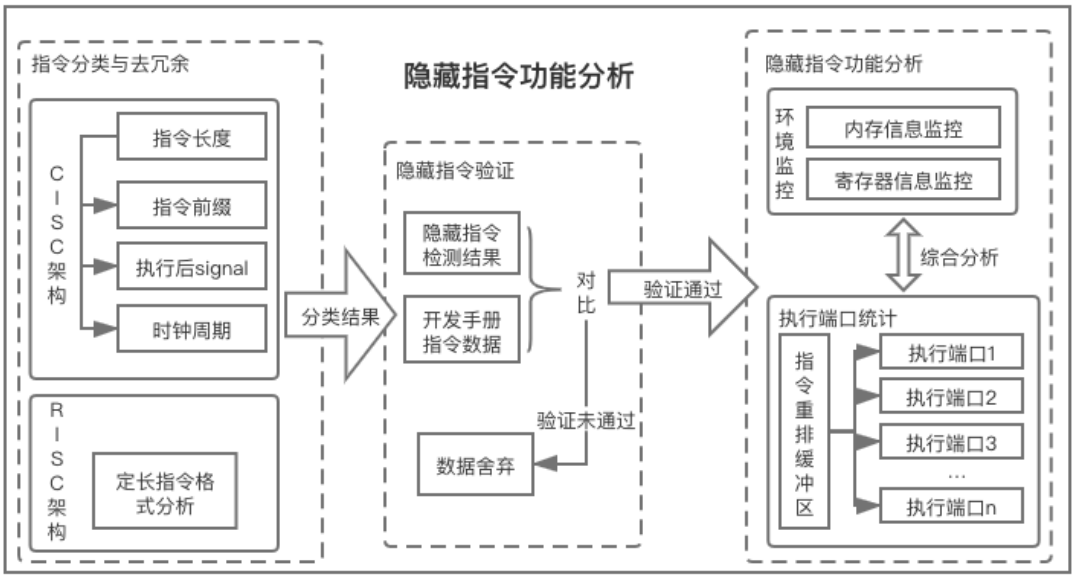

图 3：功能分析与后门检测研究内容架构图

4.项目关键问题与创新点

隐藏指令搜索是对指令集空间下的指令进行筛选的过程，是隐藏指令检测的关键所在。而部分指令集空间过大，达到 1.3x10^36 这一数量级，因此，解决隐藏指令搜索中指令空间过大的问题是提高隐藏指令检测的效率的关键所在。传统的隐藏指令搜索算法中，往往对复杂指令集（CISC）采用基于指令长度的深度优先搜索算法，对精简指令集（RISC）采用全遍历搜索的方法。由于指令格式与立即数等问题导致搜索空间过大，造成计算资源浪费，降低了隐藏指令搜索的效率。方案将采用基于指令格式分析的隐藏指令搜索算法来解决指令空间过大的问题。

- 针对 CISC 和 RISC 指令集的不同，提出了通用性的处理器芯片隐藏指令检测手段，可以在多种的处理器芯片如 x86、RISC-V、SPARC、ARM 等指令集上进行检测。
- 面对处理器芯片中指令集层面的安全，本方案提出了基于指令格式分析的隐藏指令检测算法，其极大的减小了巨大的指令搜索空间。

- 针对检测后的指令分类问题，本方案提出了一种指令分类、去冗余和验证的方法，该方法可以提升检测的准确率，并将检测结果细粒度的分类至不同的指令类别中。

5.项目应用场景

多架构隐藏指令检测及功能分析适用范围广，且可填补目前隐藏指令安全方面缺少系统化检测方式的空白。目前对于隐藏指令的检测与功能分析尚在起步阶段，国内外都缺乏成熟的系统化检测与功能分析方法，本课题中所研究的针对多架构的隐藏指令的检测与功能分析不受操作系统的限制，可用于各类型处理器平台的检测，在指令集层面为芯片提供安全保障。

6.已有的建设基础

项目团队主要从事软件/硬件/协议漏洞挖掘和利用、逆向分析技术、恶意程序检测、密码学、大数据安全分析等方面的科研工作，已发表高水平学术论文上百篇，申请专利数十项，承接了国家高科技计划 863 项目、国家重点研发计划项目、国家自然科学基金项目及联合基金项目等大量国家级项目，以及数十项省部级科研项目，已向中国国家漏洞库 CNNVD 提交百余项 0day 重大安全漏洞。基于以往科研项目的成果，已实现了针对 x86 架构隐藏指令检测的原型系统，检测了几十个 x86 架构 CPU 的隐藏指令。在本项目中，实验室将基于已有成果深入研究，将针对 x86 架构隐藏指令检测的原型系统拓展到多架构，并设计实现对于隐藏指令功能的自动化分析技术。

7.项目建设方案

7.1 建设计划

本项目计划使用 2 个月时间进行隐藏指令安全检测模块开发、3 个月时间进行隐藏指令分析模块开发、3 个月时间进行多种处理器指令集架构的安全检测、3 个月时间完成成果汇总与撰写。本项目计划在进展约 6 个月时达到复赛考核标准，包括：开发隐藏指令检测工具；制定后续工作计划。本项目计划在进展约 12

个月时达到决赛考核标准，包括：完成项目整体设计与实现工作，调试并完成演示 Demo，完成路演 PPT。

7.2 建设目标

- 处理器隐藏指令测试源代码 1 份
- 处理器隐藏指令检测测试报告 1 份
- 路演 PPT1 份