学　　号　2015301500150
密　　级

武汉大学本科毕业论文

安卓应用的动态行为捕获

院 (系) 名 称:　国家网络安全学院

专 业 名 称:　信息安全

学 生 姓 名:　蹇奇芮

指 导 教 师:　傅建明　教授

二〇一九年五月

BACHELOR'S DEGREE THESIS
OF WUHAN UNIVERSITY

Dynamic behavior capture for Android apps

School (Department): School of Cyber Science and Engineering

Major: Information Security

Candidate: QiRui Jian

Supervisor: Prof. JianMing Fu

Wuhan University

May, 2019

郑 重 声 明

本人呈交的学位论文, 是在导师的指导下, 独立进行研究工作所取得的成果, 所有数据、图片资料真实可靠. 尽我所知, 除文中已经注明引用的内容外, 本学位论文的研究成果不包含他人享有著作权的内容. 对本论文所涉及的研究工作做出贡献的其他个人和集体, 均已在文中以明确的方式标明. 本学位论文的知识产权归属于培养单位.

本人签名: _____ 日期: _____

摘　　要

智能移动终端设备的盛行使得针对移动操作系统的恶意应用迅速增加. 目前的移动操作系统中, 安卓系统巨大的市场份额和其相对开放的应用分发和权限管理方式使得其成为的攻击者的主要目标. 为了识别出恶意应用并阻止其传播, 我们需要对应用的行为进行分析. 然而单纯的静态分析在如今安卓应用成熟的混淆和加壳机制的保护下无法很好的揭示应用的行为, 因此需要动态的对安卓应用的行为进行捕获和分析.

本文分析了目前已有的一些安卓系统应用行为监测系统的实现方式和优缺点, 并且通过 hook 技术以及对安卓 8.1 源代码的修改设计和实现了一个运行于 Nexus 5x(google 的一款智能手机) 的高性能应用动态行为捕获系统. 该系统能够捕获到 Java 层的所有方法调用以及 Native 层的重要函数调用, 并且支持动态地调整需要监控的目标方法 (Java 层) 和函数 (Native 层). 本文使用常用应用对该系统进行了测试, 结果显示与同样能捕获到所有 java 层方法调用的 Android Device Monitor 相比本系统的性能开销明显更低.

本文设计思路结合了 hook 技术带来的灵活性和以及修改源代码的稳定性以及高性能, 对其他开源平台的类似工具设计有一定参考作用, 但在应用中应当注意两种方式可能的冲突问题.

关键词: 安卓应用; 动态行为; 高性能

ABSTRACT

Malicious applications on mobile operating systems boom with the prevalence of smart mobile devices. The huge market share of Android, one of current mobile operation systems, and its relatively open application distribution and privilege management make it attackers' major target. In order to identify malicious applications and prevent them from spreading, we need to analyze the behavior of applications. However, only static analysis can not handle the mature obfuscation and packing mechanism of Android applications, so it is necessary to dynamically capture and analyze the behavior of Android apps.

In this paper, I analyze the implementation, advantages and disadvantages of some existing Android application behavior monitoring systems, and present a high-performance application dynamic behavior capture system, which can run on Nexus 5x and is implemented by using hook technology and modifying Android source code. The system captures all method invocations of Java layer and important function calls of Native layer, and supports dynamic adjustment of the target methods (Java layer) and functions (Native layer) that need monitoring. I evaluate the system with common applications and the result shows that the overhead is significantly lower than that of Android Device Monitor when capturing all java layer method invocations.

The design of the system in this paper combines the flexibility brought by hook technology and the stability and high performance brought by modification of source code, which can be used for designing similar tools of other open source platforms, but we

should pay attention to the possible conflicts between the two approaches;

Key words: Android application; dynamic behavior; high performance

目　　录

摘要 III

ABSTRACT IV

1 **绪论** 1

1.1 研究背景与意义 1

1.2 国内外研究现状 1

1.3 论文主要工作 1

1.4 论文组织结构 1

2 **背景技术分析** 2

3 **系统设计实现** 3

4 **实验与结果分析** 4

5 **总结与展望** 5

1 绪论

1.1 研究背景与意义

智能移动终端设备的普及以及物联网的发展使得移动应用与人们的生产生活关系越来越密切. 这些应用一方面极大地便利了人们的生活, 提高了生产效率, 另一方面也被越来越多攻击者利用来牟取不正当利益.

1.2 国内外研究现状

1.3 论文主要工作

1.4 论文组织结构

2　背景技术分析

3　系统设计实现

4　实验与结果分析

5　总结与展望