doi:10.3969/j.issn.1000-436x.2014.z2.016

Android 平台 NFC 应用漏洞挖掘技术研究

王志强[1]，刘奇旭[2]，张玉清[2]

(1. 西安电子科技大学 综合业务网理论及关键技术国家重点实验室，陕西 西安 710071；
2. 中国科学院大学 国家计算机网络入侵防范中心，北京 101408)

摘 要：为了提高 NFC 技术的安全性，针对 Android 平台 NFC 应用进行 NDEF 协议漏洞挖掘研究，提出了一种基于 Fuzzing 技术的测试方法。该方法采用手工、生成和变异 3 种策略构造测试用例，使用报文逆向分析和嗅探 2 种手段辅助分析并构造报文；然后，利用构造的测试用例对 NFC 应用目标进行漏洞挖掘并输出结果。根据该方法，开发了一个 NFC 应用安全漏洞挖掘系统 ANDEFVulFinder，采用 logcat 和进程监控的手段在漏洞挖掘过程中对目标进行监测，并通过模拟标签和触碰操作实现漏洞挖掘过程自动化。最后，通过测试 MIUI 系统和 6 个应用，发现了 8 个漏洞，结果表明了漏洞挖掘方法的有效性。

关键词：近场通信；移动设备；Fuzzing 技术；漏洞挖掘

中图分类号：TP309.2 **文献标识码**：A **文章编号**：1000-436X(2014)Z2-0117-07

Research of discovering vulnerabilities of NFC applications on Android platform

WANG Zhi-qiang[1], LIU Qi-xu[2], ZHANG Yu-qing[2]

(1. State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China;
2. National Computer Network Intrusion Protection Center, University of Chinese Academy of Sciences, Beijing 101408, China)

Abstract: To improve the security of NFC technology, a research is done for discovering NDEF vulnerabilities of NFC applications on Android platform, and a method of bug hunting is proposed on based Fuzzing technology. The method adopts manual craft, the generation and the mutation strategies to construct test cases, and uses two assistant means of analyzing and constructing test cases, including reverse message anylysis and packet sniffing. Then, NFC applications' vulnerabilities with constructed test cases and output results are discovered. According to the method, a system called ANDEFVulFinder is developed for discovering the security vulnerabilities of NFC applications. The tool logcat and process monitoring are used to monitor targets' exceptions during the discovering process, and the test is automated achieved by tag emulation and "touch" operation emulation. Finally, 8 vulnerabilities are found by doing lots of experiments on MIUI operating system and 6 NFC applications, which has proved proposed method's effectiveness.

Key words: near field communication; mobile devices; Fuzzing; vulnerability discovering

1 引言

NFC(near field communication)技术是一种近距离的双向高频无线通信技术，能够在移动终端、智能标签(tag)等设备间进行非接触式数据交换[1,2]。NFC 技术具有通信距离短、一次只和一台设备连接、硬件安全模块加密等特点，具有较好的保密性和安全性。由于 NFC 技术的迅速发展，其安全性受到越来越多的威胁和挑战。如何有效地挖掘并修复 NFC 安全漏洞，是 NFC 技术安全亟待解决的关

收稿日期：2014-05-08

基金项目：国家自然科学基金资助项目（61272481，61303239）；国家发改委信息安全专项基金资助项目（发改办高技[2012]1424）

Foundation Items: The National Natural Science Foundation of China (61272481，61303239); The National Development and Reform Commission Special Notice of Information Security([2012]1424)

键问题。

目前研究人员针对 NFC 应用安全漏洞挖掘和防护策略已经做了大量的研究。文献[3]提出一种手动的 NFC 手机安全漏洞挖掘方法，通过把报文写入标签并使用 NFC 手机触碰测试，发现了标签内容欺骗、URL 钓鱼、应用 DoS 等漏洞。该方法采用手工构造测试用例进行测试，无法实现自动化，耗费人力和时间，效率低下。文献[4]使用基于生成和变异策略结合的 Fuzzing 技术对协议层和应用层测试，利用 Sulley 完成变异测试，同时使用 logcat 进行监控异常，发现了大量的漏洞。该方法虽然采用基于生成和变异结合的方法构造测试用例，但针对具体报文时却采用单一的策略构造测试用例，此外，针对目标异常的监控手段也比较单一。文献[5]开发一个测试框架，针对 Android NFC API 和 NFC APP 进行测试，使用 Sulley 生成测试用例，利用 Intent 发送报文，对 7 个应用进行测试，发现了一些 DoS 漏洞。该方法采用 Sulley 的进程监控方式，对智能手机的监控效果不好，且监控手段比较单一，不能提供详细的 logcat 异常日志；同时，该方法对 Android 系统和 API 版本依赖强，可移植性差。文献[6]设计并开发了小型无源的“补丁”EnGarde，可嵌入手机后壳，用于拦截 NFC 恶意交互，包括拦截恶意操作、智能拦截黑名单中的行为事件等。该方法是从防护的角度对 NFC 安全进行研究，需要增加硬件成本。

针对当前 NFC 漏洞挖掘工作存在的问题，提出一种基于 Fuzzing 技术的 NFC 安全漏洞挖掘方法。首先，根据已知漏洞和报文格式，手工构造测试用例并进行测试；其次，根据报文格式，采用基于生成的变异策略构造测试用例并进行测试；最后，把测试出的异常数据和已知漏洞数据作为输入，使用基于变异的策略构造测试用例，对目标测试和输出结果。同时，为了提高测试用例构造的效率，采用数据分组逆向和数据分组嗅探 2 种方法获取并还原数据分组，用于辅助分析和构造数据报文。

基于上述测试方法，开发了一个 NFC 应用安全漏洞挖掘系统 ANDEFVulFinder，采用 logcat 和进程监控的方式检测目标，使用 ACR ACS 122U 模拟标签，通过进程控制命令模拟触碰操作，实现了测试的自动化。最后，针对 MIUI 系统(MIUI-4.1.17)和 6 个 NFC 应用进行漏洞挖掘，发现了大量的漏洞，结果表明了工具的有效性。

2 NFC 应用漏洞挖掘方法

2.1 NDEF 协议

NFC 应用使用的数据交换格式是 NDEF 协议，即 NFC 数据交换格式(NFC data exchange format)，它是一种 NFC 设备和 NFC 标签使用的普通数据报文格式[7]，用于 NFC 设备之间、NFC 设备与标签之间交换具体的应用或者服务数据，实现互联互通的功能。NDEF 是一种轻量级二进制报文格式，在单个报文结构中可封装多个数据载荷(payload)。每个 NDEF 报文包含一个或者多个 NDEF 记录(record)，每个记录可以包含任意数据类型长度最大为 $2^{31}-1$ byte 的载荷。具体记录的格式和各个字段的介绍请参考 NFC Forum 官方文档[7]。

2.2 漏洞挖掘方法

针对 NFC 数据交换格式 NDEF，采用 Fuzzing 技术构造测试用例进行漏洞挖掘。Fuzzing 技术是一种自动化软件测试技术，通过向被测目标输入大量的畸形数据并监测其异常来发现漏洞，是漏洞挖掘的重要手段之一[8]。

漏洞挖掘方法可以分为 3 个阶段：手工测试、使用生成策略和变异策略构造测试用例并测试。手工测试可以辅助分析、构造测试用例，提高测试用例的有效性，有利于弥补基于生成和变异无法生成某些特殊测试用例的缺陷；生成和变异的策略是 Fuzzing 技术中构造测试用例的常用方法，两者结合能够互相弥补各自缺点，并自动化生成大量测试用例。

研究中使用的漏洞挖掘方法可描述如下。

Step1 初始化，为 3 个阶段的测试做准备；

Step2 使用手工构造测试用例对目标测试，并得到异常数据；

Step3 使用生成策略构造测试用例对目标测试，并得到异常数据；

Step4 使用得到的异常数据作为样本，使用变异方法构造测试用例，并进行测试和监控；

Step5 输出测试结果和日志。

关于初始化、手工分析和测试、构造测试用例、监控等具体细节将在下一节具体实现时介绍。

3 ANDEFVulFinder 系统架构和实现

3.1 系统架构

根据第 2 节 NFC 漏洞挖掘方法，在 libnfc 开源

库的基础上，使用 C 语言开发了一个针对 Android 平台的 NFC 应用安全漏洞挖掘系统 ANDEFVulFinder，系统架构如图 1 所示。NFC 漏洞挖掘系统框架分为 6 部分：测试用例构造、手机初始化、NFC tag 测试、异常监控、异常验证、日志输出。

1) 测试用例构造模块用于构造测试报文，报文类型包括 NFC 论坛类型、NFC 论坛外部类型、绝对 URI 类型和 MIME 类型等，构造策略将在 4.2 节描述。

2) 手机初始化模块用于卸载无关目标，排除其他 NFC 应用程序对目标的干扰，同时安装测试目标到手机上，为测试做准备。

3) NFC tag 测试模块包括 tag 模拟和触碰模拟，用于模拟各种标签和模拟手机触碰标签的过程，实现测试的自动化；其中标签模拟是通过 libnfc 库向 NFC 设备上写入数据报文实现的，而触碰操作的模拟手段是通过向手机发送 kill 命令，控制手机进程 com.android.nfc 的停止和运行。

4) 异常监控模块采用 logcat 和进程监控 2 种手段监控 NFC 手机系统和应用程序，并记录异常状态下的测试用例和目标的断点状态，为异常分析和重现做准备。

5) 异常验证模块针对发现的异常，通过重新发送测试用例和监视异常对漏洞进行确认。

6) 日志输出模块输出测试结果。

3.2　测试用例构造

测试用例构造采用的策略分为 3 个阶段：手工构造、基于生成的多维构造策略、基于变异的多维构造策略。

1) 手工构造

该方法是指通过分析 NFC 论坛官方文档和 NDEF 漏洞，手工构造测试用例并对目标进行测试。手工构造方式不仅可以提高畸形测试用例的有效性，还能够重现已知漏洞和发现新漏洞。关于已知漏洞的分析如下。

CVE-2008-5825：手机 Nokia 6131(固件为 05.12)的智能海报应用遇到包含空格(0x20)、回车(0x0D)或点(0x2E)字符的 URI 记录时，无法正常显示该记录。攻击者通过构造包含以上畸形数据的 URI 记录，可以诱导用户访问恶意网站、拨打收费的电话或发送购买彩铃的短信。

CVE-2008-5826：通过把 NDEF 记录的载荷长度、NDEF URI 电话记录的长度字段或 NDEF URI 短信记录的长度字段设置为一个大值并写入标签，会导致手机 Nokia 6131(固件为 05.12)在触碰该类标签时崩溃。

CVE-2008-5827：手机 Nokia 6131(固件为 05.12)在下载完 JAR 文件时会自动安装软件，远程攻击者可以利用此特性构造 URI 记录执行任意代码。

根据已知漏洞，手工构造包含空格、回车、点、畸形长度字段等的畸形报文和文件传输报文进行测试。

2) 基于生成的多维构造策略

该策略是指通过分析 NFC 论坛官方文档，获取各类 NDEF 报文的格式，使用畸形数据生成 NDEF 报文的一个或多个测试字段，其他字段使用正常数据填充。针对各类报文的头、长度字段、类型字段、载荷等字段，构造了一个畸形数据库，如表 1 所示，包括整型值、字符串、目录遍历、分隔符和其他非字母字符等。例如，在构造包含 URL 链接的智能海报报文过程中，使用整型值生成 URL 记录的载荷长度字段，使用字符串和分隔符生成 URL 字段。

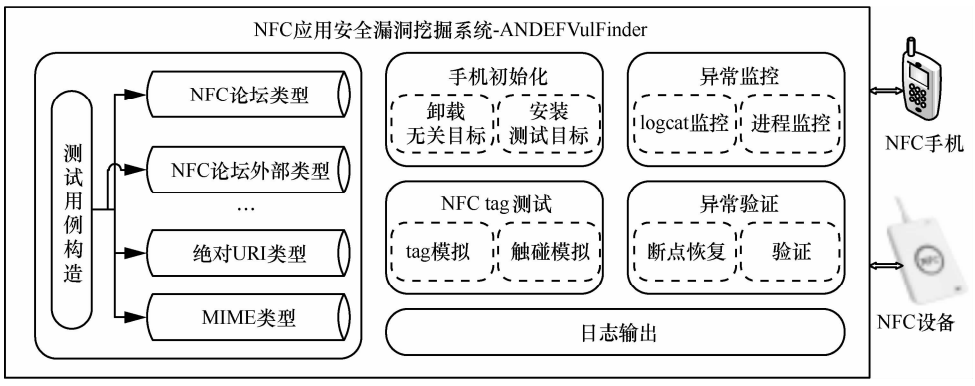


图 1　ANDEFVulFinder 系统架构

表 1 畸形数据库

类型	值
整型值	零值、负值与边界值：0x00,0x01,0x7F,0x80,0xFE,0xFF; 0x0000, 0x0001,0x7FFF,0x8000,0xFFFE,0xFFFF
字符串	长字符串：AAA…, BBB…; 格式化字符串：%s, %n, %s, %x, %n%s,…
目录遍历	~/,/…,../../,\.,\..,..\..\,…
分隔符及其他	0x0A, 0x1C~0x1F, 0x20~0x2F, 0x3A~0x40, 0x5B~0x60, 0x7B~0x7E

3) 基于变异的多维构造策略

该策略不需分析 NFC 论坛文档，根据给定的报文样本，对样本中的一个或多个字段采用随机变异的方式构造测试用例。由于历史上导致漏洞的样本数据可能会再次触发漏洞，因此，选择已知漏洞数据和前 2 个阶段导致异常的测试用例作为样本。例如，给定一个包含 URL 链接的智能海报样本，对 URL 记录的长度字段进行比特变异操作，构造出新的测试用例。

由于有些 NFC 应用识别的数据报文略有差异，例如云飞应用构造的标签，使用 MIUI 系统则无法识别，系统会自动打开云飞的官网，提示下载和安装云飞应用。为了增大以上 3 种方法构造测试用例的识别率，采用逆向分析数据分组和嗅探的方法，用来辅助构造测试用例，下面将分别介绍 2 种方法。

逆向分析数据分组的方法如下。

a) 使用 Android 平台 NFC 应用程序构造智能海报、URL 打开、发送短信、蓝牙配对、WiFi 连接等报文；例如 Detail!、TagInfo、TagWriter 等图形界面应用，只需要输入 URL、短信、蓝牙地址、WiFi SSID 和密码等即可。

b) 通过 NFC 手机触碰 NFC 标签，把构造好的报文写入标签中。

c) 使用具有报文分析功能的 NFC 应用读取标签中的内容，存储到手机存储卡。例如，TagInfo 应用可以读取标签中内容，并以二进制方式存储到手机存储卡中。

d) 使用 adb pull 命令把报文的二进制文件导出到计算机中。

嗅探数据分组的方法如下。

a) 使用手工或者基于生成的策略构造报文，并把报文写入 NFC 标签中。

b) 使用 NFC 读卡器读取标签内容，例如 ACS ACR122U、Proxmark3、SCL3711 等读卡器。

c) 在上一过程中，使用嗅探器 Proxmark3 嗅探通信数据。

d) 分析嗅探到的通信数据，还原出数据分组。

通过把二进制文件或还原出的数据分组写入读卡器，即可利用卡模拟模式虚拟一个 NFC 标签。由于二进制文件或还原的数据分组完全是可被完全识别的通信数据，虚拟的标签也可以完全被 NFC 系统应用和第三方应用所识别。因此，通过对二进制文件或还原的数据分组进行分析，可以提高 3 种策略构造方法的效率。

3.3 测试与监控

实现自动化测试的 2 个核心问题是标签模拟和触碰操作模拟。标签模拟可以通过 NFC 设备实现，该类设备包括 ACS ACR122U、Proxmark3 等。因为 ACS ACR122U 是基于开源库 libnfc 的，使用 C 语言编写，易于模拟 NFC 标签和系统移植，所以选择该设备作为标签模拟器。触碰操作是指 NFC 设备从远靠近标签的操作，该操作可以通过命令 kill -s SIGSTOP PID 和命令 kill -s SIGCONT PID 控制 NFC 进程的暂停和恢复来模拟。一些 NFC 应用在标签变化时会自动读取标签内容，对于该类标签不需触碰模拟操作，即不需使用 kill 命令进行操作。

针对 NFC 应用异常的监控采用 2 种手段：logcat 监控和进程监控。命令 logcat 有多种监控选项和过滤操作，能够输出详细的应用活动日志，包括警告、错误等，是 Android 应用监控和调试的重要工具。进程监控采用 top 命令实现，该命令可以监控 CPU 使用率、进程状态、内存占用等情况。

4 实验配置和环境

实验配置分为硬件配置和软件配置。硬件配置是指 ANDEFVulFinder 运行依赖的硬件设备及其相关信息，软件配置为测试的目标及其测试环境设置等。

硬件配置：ANDEFVulFinder 运行于一台配置为 Intel(R) Core(TM) i7 单处理器 2.93 GHz、内存 512 MB、系统版本为 Linux ubuntu 2.6.32-21-generic 的虚拟机下。NFC 手机配置为系统版本为 MIUI-4.1.17 (Android 版本为 4.1.1 JRO03C)、处理器为四核 1.4 GHz、内存为 1 GB 的三星 GT-I9300 手机。NFC 设备为符合 ISO/IEC 18092 标准的 ACS ACR122U 智能卡读写器，也用于数据分组逆向分析。数据分组的嗅探工作使用 Proxmark3 工具进行完成。

软件设置：指选择的测试目标及其信息，如表

2 所示，包含一个系统进程和 6 个 NFC 应用程序。由于 NFC 应用相互之间会产生冲突，因此，在测试前需要使用 ps 命令找出测试目标以外的所有 NFC 应用分组名，并使用 adb uninstall 卸载掉这些应用。同时使用 adb install 命令把被测目标安装到手机系统中。

表 2　　测试目标

测试目标	版本	开发者
MIUI	4.1.17	谷歌/小米
TagInfo	2.00	NXP 公司
TagWriter	2.3	NXP 公司
NFC 标签助手	1.05	上海复旦微电子集团
NFC Detail!	0.5	U&I Reseach Lab
小木公交	3.0	Share More Studio
NFC Developer	2.1.1	Thomas Rorvik Skjolberg

根据以上软硬件设置，搭配实验环境如图 2 所示，其中图 2(a)为数据分组逆向分析实验环境，分析步骤分为①写入内容；②读取标签；③导出二进制文件。图 2(b)为数据分组嗅探环境，嗅探步骤分为①NFC 手机触碰标签；②proxmark3 修改通信内容；③导出并还原数据分组。图 2(c)为 ANDEFVulFinder 测试系统的环境搭配，首先利用 ACR122U 虚拟标签，然后使用手机触碰虚拟标签，其次利用 logcat 和进程监控方式监测异常，循环测试直至结束。

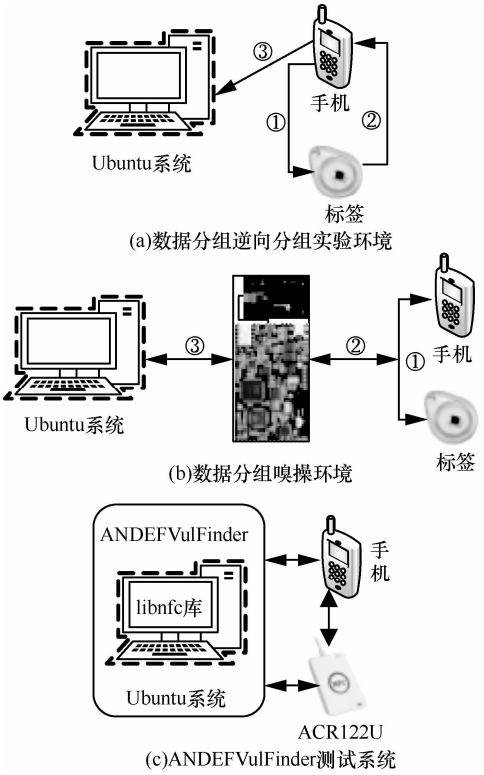


图 2　实验环境

5　实验结果

能否挖掘到未公布的安全漏洞，是衡量漏洞挖掘工具是否有效的最重要的标准。使用 ANDEFVulFinder 对目标测试后，成功发现一些未公开的系统和应用漏洞，包括 NFC 服务崩溃漏洞、自动打开手电筒漏洞、自动打开蓝牙和 WiFi 漏洞、NFC 进程僵死、第三方应用拒绝服务、浏览器跳转漏洞等，导致这些漏洞原因有畸形的字段长度、设计逻辑错误、能量消耗、报文解析错误、未过滤特殊字符等。下面将具体描述各种漏洞。

1) MIUI 系统 NFC 服务崩溃

该漏洞属于拒绝服务漏洞，导致 NFC 服务崩溃的原因是在包含蓝牙配对记录的 NDEF 报文中，本地名字段与字段长度不匹配导致的，危害等级为低危。该类漏洞分为 2 类：本地名字段长度为零触发的异常和长度为负值触发的异常，即该长度字段为 0b0000 0000 或者 0b1xxx xxxx。

2) MIUI 系统自动打开手电筒

该漏洞属于设计缺陷，通过构造分组名为 com.android.systemui 的 pkg 分组启动报文，可导致 MIUI 系统在触碰该标签时自动打开系统手电筒，危害等级为中危。com.android.systemui 分组对应的系统应用时 MiuiSystemUI.apk，该程序是 MIUI 系统的一个核心应用程序，具有电量管理、快捷键设置、系统设置、状态栏、任务管理等多种功能，反编译后发现手电筒 TorchActivity 设置为根目录下唯一后缀名为 Activity 程序，导致手电筒被自动打开。

3) MIUI 系统自动打开蓝牙

该漏洞属于逻辑漏洞，使用安装 MIUI 系统的 NFC 手机触碰任意正常的蓝牙配对报文，均会导致蓝牙被打开，危害等级为高危。正常的逻辑操作是：如果蓝牙处于打开状态，不管配对成功与否均会保持蓝牙的打开状态；如果蓝牙处于关闭状态，配对成功则会打开蓝牙，否则自动关闭蓝牙开关。该漏洞的效果如图 3 所示。

4) MIUI 系统自动打开 WiFi

该漏洞也属于逻辑漏洞。实验中的 MIUI 系统不能识别 WiFi 连接报文，需要安装系统插件 WifiHandover 或 NFC 标签助手等应用协助连接 WiFi，危害等级为中危。使用手机触碰包含 WiFi

连接报文的标签后，系统 WiFi 会被自动打开，效果和自动打开蓝牙漏洞类似。

图 3 打开蓝牙漏洞

5) MIUI 系统 NFC 进程僵死

该漏洞与 NDEF 消息报文格式关系不大，其触发原因是某些标签能量消耗过大，超过了手机 NFC 芯片的供电能力，导致手机 NFC 进程僵死，危害等级为低危。使用测试手机不断触碰嵌套或包含多个纪录的 NDEF 报文，手机进程 com.android.nfc 变成停止状态“S”，ACS ACR 122U 指示灯由橘黄色变成红色，正常的卡模拟模式失效。

6) TagInfo 应用崩溃

该漏洞属于拒绝服务漏洞，危害等级为低危。由于 TagInfo 应用无法正确解析部分 NDEF 报文，包括签名报文、智能海报、文本、URI 报文等，NFC 手机触碰该类报文时，TagInfo 应用提示 java.lang.IllegalStateException 异常。

7) Detail!崩溃

该漏洞属于拒绝服务漏洞，危害等级为低危。由于 Detail!应用无法正确解析部分 NDEF 报文，包括智能海报 URL 报文、WiFi 连接报文等，使用 NFC 手机触碰写入该类报文的标签时，Detail!发生崩溃，logcat 输出 java.lang.IndexOutOfBoundsException 异常。

8) MIUI 默认浏览器 URL 欺骗漏洞

该漏洞属于 MIUI 默认浏览器的漏洞，危害等级为中危，主要是通过默认浏览器打开标签中的畸形 URL 导致的。使用 NFC 手机触碰 URL 为 11.com@22.com 的 URI 报文，默认浏览器会自动跳到 22.com，这是由于默认浏览器未过滤特殊字符导致，攻击者可以利用该漏洞进行钓鱼攻击和诈骗。

6 对比与解决措施

本节首先从测试用例构造、监控方式、移植性、自动化程度、漏洞情况等 5 个方面与相关工作进行对比和分析，其次针对发现的漏洞提出了响应的解决措施和修复方案。

6.1 对比

为了说明 ANDEFVulFinder 的有效性和价值，搜集了近年来 NFC 漏洞挖掘和安全防护的相关工作进行比较，对比结果如表 3 所示，分析如下。

1) ANDEFVulFinder 在测试用例构造方面更加细致，增加基于已知漏洞知识和手工构造，更加有利于发现漏洞。

2) 在监控方式方面，ANDEFVulFinder 比其他 3 个文献更加全面，可以监控崩溃日志、CPU 使用率、进程运行状态等。

3) ANDEFVulFinder 和文献[4]的移植性比较好；文献[3]采用手工测试，不存在测试工具或系统的移植性问题；文献[5]对需要依赖具体的 Android 平台，移植性比较差。

4) 除了文献[3]外，ANDEFVulFinder 和文献[4,5]测试的自动化程度比较高。

5) 由于 ANDEFVulFinder 依据已知漏洞知识和人工手段构造测试用例，因此，可以覆盖文献[3~5]中的所有漏洞。此外，ANDEFVulFinder 还发现了自动打开手电筒、能量消耗导致的进程僵死、自动打开蓝牙和 WiFi 等新漏洞。

表 3 对比结果

方案	测试用例构造	监控手段	移植性	自动化	漏洞
文献[3]	手工构造	人工	—	—	标签内容欺骗、URL 欺骗、NFC 应用拒绝服务等漏洞
文献[4]	采用基于生成或基于变异的策略	logcat 监控	好	高	NFC Service 拒绝服务、URL 欺骗、蓝牙配对等漏洞
文献[5]	基于生成的策略	Sulley 进程监控	差	高	NFC Service 拒绝服务、TagInfo 等应用拒绝服务漏洞
ANDEFVulFinder	手工构造、基于生成和基于变异的策略	Logcat 监控和 top 进程监控	好	高	NFC Service 拒绝服务、URL 欺骗、NFC 应用的拒绝服务、蓝牙配对、NFC 设计缺陷等漏洞

6.2　解决措施

针对以上漏洞，提出相应的解决措施和建议如下，供 Android 系统和应用开发者参考。

1) 检查长度字段的零值和负值，采取限制字段长度或把转换的方法识别正确的长度；

2) 修改蓝牙配对、WiFi 连接的 NFC 进程的设计逻辑，当配对失败或连接失败时恢复原始状态；

3) 修改 MIUISystemUI.apk 的设计逻辑，禁止启用 TorchActivity 或者设置默认的启动程序；

4) 对于能量消耗过大的标签，例如，包含长 title 或者嵌套记录的智能海报报文，增加对标签长度和嵌套次数的限制；

5) 严格按照 NFC 论坛官方文档编写 NFC 应用程序，避免 NFC 应用的拒绝服务漏洞；

6) 增加对标签欺骗报文中特殊字符的过滤，防止通过 NFC URL 报文下载恶意软件并安装。

7　结束语

本文提出一种基于 Fuzzing 技术的 NFC 应用安全漏洞挖掘方法，开发了一个 NFC 应用安全漏洞挖掘系统 ANDEFVulFinder。该系统采用手工构造、基于生成和基于变异的三级策略构造测试用例，使用数据分组逆向分析和数据分组嗅探 2 种手段辅助分析和构造测试用例，使用 logcat 和进程监控方法监测目标。通过模拟标签和触碰操作实现了自动化的测试，并针对 MIUI 系统和 6 个 NFC 应用进行测试，发现了大量的未知漏洞。

参考文献：

[1] MADLMAYR G, KANTNER C, GRECHENIG T. Secure Smart Embedded Devices, Platforms and Applications[M]. New York: Springer, 2014: 351-367.

[2] COSKUN V, OZDENIZCI B, OK K. A survey on near field communication (NFC) technology[J]. Wireless Personal Communications, 2013, 71(3): 2259-2294.

[3] MULLINER C. Vulnerability analysis and attacks on NFC-enabled mobile phones[A]. Proceedings of the 2009 IEEE International Conference on Availability, Reliability and Security(ARES'09)[C]. Fukuoka, Japan, 2009.695-700.

[4] MILLER C. Exploring the NFC attack surface[EB/OL]. http://media.blackhat.com/bh-us-12/Briefings/C_Miller/BH_US_12_Miller_NFC_attack_surface_WP.pdf, 2012.

[5] WIEDERMANN N. Fuzzing-to-go:A test framework for Android devices[D]. Technische Universität München, 2012.

[6] GUMMESON J J, PRIYANTHA B, GANESAN D, *et al*. EnGarde: Protecting the mobile phone from malicious NFC interactions[A]. Proceeding of the 11th ACM annual international conference on Mobile systems, applications, and services(MobiSys'13)[C]. Taipei, China, 2013.445-458.

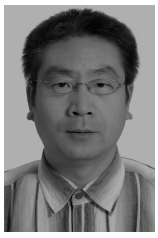
[7] NFC Forum. NFC Data Exchange Format (NDEF) Technical Specification[S]. 2006.

[8] SUTTON M, GREENE A, AMINI P. Fuzzing: brute force vulnerability discovery[M]. New Jersey: Pearson Education, 2007.

作者简介：

王志强（1985-），男，安徽宿州人，西安电子科技大学博士生，主要研究方向为漏洞挖掘。

刘奇旭（1984-），男，江苏徐州人，博士，中国科学院大学讲师，主要研究方向为漏洞挖掘与漏洞评估。



张玉清（1966-），男，陕西宝鸡人，中国科学院大学教授、博士生导师，主要研究方向为密码学、网络与信息系统安全。