软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2018,29(5):1392−1409 [doi: 10.13328/j.cnki.jos.005494]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于宿主权限的移动广告漏洞攻击技术*

王持恒[1], 陈 晶[1], 苏 涵[1], 何 琨[1], 杜瑞颖[1,2]

[1](国家网络安全学院(武汉大学),湖北 武汉 430072)

[2](地球空间信息技术协同创新中心,湖北 武汉 430079)

通讯作者: 陈晶, E-mail: chenjing@whu.edu.cn

摘 要: 移动广告作为市场营销的一种重要手段,越来越受到应用开发者的青睐,其市场规模也日趋增大.但是,为了追求广告的精准投放和其他非法利益,移动广告给用户的隐私与财产安全也带来了很大的威胁.目前,众多学者关注广告平台、广告主和移动应用的安全风险,还未出现在广告网络中直接发起攻击的案例.提出了一种基于宿主权限的移动广告漏洞攻击方法,能够在移动应用获取广告内容时,在流量中植入攻击代码.通过对广告流量的拦截,提取出宿主应用的标识和客户端相关信息,间接得到宿主应用的权限列表和当前设备的 WebView 漏洞.另外,提出了一种攻击者的能力描述语言,能够自动生成定制化的攻击载荷.实验结果表明,所提出的攻击方法能够影响到大量含有移动广告的应用.几个攻击实例的分析也证明了自动生成攻击载荷的可行性.最后,提出了几种防护方法和安全增强措施,包括应用标识混淆、完整性校验和中间人攻击防护技术等.

关键词: 移动广告生态系统;宿主权限;中间人攻击;攻击载荷自动生成;能力描述语言

中图法分类号: TP309

中文引用格式: 王持恒,陈晶,苏涵,何琨,杜瑞颖.基于宿主权限的移动广告漏洞攻击技术.软件学报,2018,29(5):1392−1409. http://www.jos.org.cn/1000-9825/5494.htm

英文引用格式: Wang CH, Chen J, Su H, He K, Du RY. Mobile advertising loophole attack technology based on host app's permissions. Ruan Jian Xue Bao/Journal of Software, 2018,29(5):1392−1409 (in Chinese). http://www.jos.org.cn/1000-9825/5494.htm

Mobile Advertising Loophole Attack Technology Based on Host APP's Permissions

WANG Chi-Heng[1], CHEN Jing[1], SU Han[1], HE Kun[1], DU Rui-Ying[1,2]

[1](School of Cyber Science and Engineering (Wuhan University), Wuhan 430072, China)

[2](Collaborative Innovation Center of Geospatial Technology, Wuhan 430079, China)

Abstract: As an important channel for mobile marketing, mobile advertising has become more and more popular among app developers. However, in pursuit of targeted ads delivery and other illegal tactics, mobile ads may introduce serious threat to users' privacy and property. Recently, many researches have paid attention on the threat of advertisement platforms, advertisement providers, and mobile apps, though few studies put focus on the security of advertisement network. In this paper, based on the automatic analysis of host app's permissions, a man-in-the-middle (MITM) attack scheme is proposed to inject malicious code into the ads' traffic. Through analyzing network traffic, this method can identify the name of host app and extract the permissions from the official app market. Moreover, it also extracts the device information such as system version and sensors, which is helpful to excavate the loophole of corresponding WebView. To generate the attack code automatically, a capability description language (CDL), which can describe the attacker's ability in a

* 基金项目: 国家自然科学基金(61572380, 61772383, 61702379); 国家重点基础研究发展计划(973)(2014CB340600)

Foundation item: National Natural Science Foundation of China (61572380, 61772383, 61702379); National Program on Key Basic Research Project (973) (2014CB340600)

本文由软件安全漏洞检测专题特约编辑王林章教授、陈恺研究员、王戟教授推荐.

收稿时间: 2017-06-22; 修改时间: 2017-08-29; 采用时间: 2017-11-21; jos 在线出版时间: 2018-01-09

CNKI 网络优先出版: 2018-01-11 17:24:35, http://kns.cnki.net/kcms/detail/11.2560.TP.20180111.1724.005.html

standardized format, is also developed. The distribution of loopholes among different Android versions are studied. Experimental results show that the proposed attack scheme can affect many apps, and the attack cases also illustrate the feasibility of this work. In the end, several protection methods and security enhance schemes, including host app name confusion, ads content integrity check, and the remission technologies of MITM attacks, are put forward.

Key words: mobile advertising ecosystem; host permission; man-in-the-middle attack; automatic attack code generation; capabilit description language

随着信息技术的飞速发展,移动互联网、云计算、社交网络等各种新技术、新应用层出不穷,移动应用广告也越来越受到开发者与品牌商的青睐.据 Google 发布的数据显示[1],2016 年,全球数字广告市场规模达 2 000 亿美元,其中,移动广告规模达到 1 000 亿美元,占数字广告总额的 50%.手机、互联网等移动媒体与传统平面媒体相比,具有实时在线、随身携带、情景感知的特点,使得其可以提供大量的个性化服务.因此,相对于传统广告形式,移动应用广告在精准匹配、互动性等方面具有明显的优势.移动应用广告可以根据用户的实际情况和实时情景将广告直接推送到用户的手机上,真正实现了“精致传播”.

移动广告的快速增长主要归因于新技术的兴起,包括定向和追踪技术等.尤其是定向技术的创新发展,实现了移动广告的精准营销,有越来越多的企业选择使用移动设备推送广告.现有的定向技术有很多,例如设备定向、内容定向、行为定向、人群特征定向、位置定向和地域定向等.所有这些定向信息使得每个用户的特征更加独特,能够显著表明用户的行为习惯和个性化需求.对于广告主来说,如何确定广告投放的有效性是最重要的问题.移动应用广告作为继应用商店后的新型商业模式,试图让广告主的每一次投放都带来利益.

移动广告生态系统主要包括 4 个实体:广告主、广告平台、移动应用和用户,如图 1 所示.其中,广告主是产品的制造者和内容的生成者,例如可口可乐、汽车厂商和零售商店等;广告平台是广告内容的投放者和管理者,包括客户端 AdSDK 库和服务器端,例如 Google 官方提供的 AdMob,国内的亿动广告、百分通联和多盟等;移动应用是广告的展示者,本文称为宿主应用,例如美团、新浪微博和今日头条等;用户就是观看广告的人,是广告最终的点击者.目前,越来越多的移动应用倾向于免费提供服务,内置广告成为应用开发者重要的盈利模式.如果用户在使用移动应用的过程中点击了广告,那么广告平台就需要支付一定的费用给应用开发者.而每次点击有可能会给广告主带来直接的利润,所以广告主需要给广告平台支付一定的投放费用.

Fig.1 Mobile advertising ecosystem

图 1 移动广告生态系统

理想情况下,整个移动生态系统能够完美配合,各方都能达到自己的利益和需求.然而,在各种利益链的驱动下,现实情况却并非如此.除了用户自己之外,其他 3 个实体都可能是不可信的.例如,广告平台为了广告内容的精准投放,可能会通过广告开发包 AdSDK 收集大量的用户数据并反馈给广告主,包括用户的系统设置、地理位置、网络状态,甚至通讯录、历史记录和应用安装列表等.恶意的广告主本身也可能在发布的广告内容中加入攻击代码,以窃取用户的隐私和破坏系统.另外,应用开发者为了带来大量的广告收益,可能会采取模拟点击、自动执行的办法欺骗广告平台,而且也能从定制化的广告中间接地推测出用户的隐私.所有这些攻击方式给用

户安全带来了极大的威胁,某些严重的隐私数据(例如联系人、短信等)甚至可能会给受害者带来财产损失.

本文在假设广告主、广告平台和移动应用都是可信的前提下,提出了一种针对 Android 移动广告漏洞的定制化攻击方法,能够在广告流量中,根据宿主权限和系统漏洞自动生成可行的攻击载荷.首先,通过对广告流量的分析,提取出宿主应用的标识,并获得当前设备的相关信息;然后,根据宿主应用的标识从应用市场中获取其权限列表,确定攻击者的能力上限;接下来,根据设备相关信息能够分析客户端的 WebView 隔离漏洞,从而确定载荷的攻击途径;最后,根据宿主权限和框架漏洞,提出一种攻击者的能力描述语言,能够自动生成定制化的攻击载荷.该方法能够合理利用已有的资源发动攻击,隐蔽性更强,有效地避免了攻击能力受限时被用户察觉,无法通过程序分析的方法对移动应用的安全性进行审查.另外,本文也提出了一些防护措施和安全增强方案来抵御这种新的攻击方式.

本文第 1 节对移动生系统的背景知识进行介绍,包括研究现状和威胁模型.第 2 节详细介绍本文所提出的攻击方法和步骤.第 3 节重点阐述基于宿主权限的载荷自动生成方法和攻击方案的具体实现流程.第 4 节通过实验与分析验证所提出的攻击方法的可行性.第 5 节给出了一些防御措施.第 6 节总结本文的研究工作.

1 背 景

1.1 移动广告生态系统

近年来,随着移动广告行业的飞速发展,针对移动广告生态系统的攻击也越来越多.正常情况下,移动广告对某些用户数据与设备信息的访问是可以理解的,精准的广告投放与广告效果监测会推动整个移动广告生态系统良性发展.但是,过度的、甚至是恶意的用户数据获取也能够给用户的隐私与安全带来严重威胁.目前,研究人员从广告主、广告平台和移动应用这 3 个角度提出了一些攻击方案和防御方法[2−14],这为移动广告生态系统的快速扩张敲响了警钟.本文对这些攻击和防御方法进行了总结,见表 1.

Table 1 Research status of mobile advertising ecosystem

表 1 移动广告生态系统研究现状

研究方向	攻击方法			防御方法
	广告平台	广告主	移动应用	
典型论文	Demetriou S, *et al*.[2] Meng W, *et al*.[3] Book T, *et al*.[4] Grace M, *et al*.[5] Datta A, *et al*.[6]	Son S, *et al*.[7] AdJail[8] Zarras A, *et al*.[9] Li Z, *et al*.[10]	Meng W, *et al*.[3] Decaf[11] Madfraud[12]	Demetriou S, *et al*.[2] Adjail[8] Draco[13] AdDroid[14] AdSplit[15]

在广告平台方面,作为广告主和移动应用之间的纽带,为了广告投放的精准度,它们会主动地收集大量用户数据,例如位置、使用习惯和兴趣爱好等[2−6].广告平台收集用户信息主要是通过嵌入在宿主应用中的 AdSDK 库实现.由于宿主应用和广告库之间没有很好的隔离,宿主应用具有的权限,广告平台同样具有,而且用户很难分辨权限请求是广告库发起的,还是正在使用的移动应用发起的.例如,文献[2]深入研究了广告库窃取用户隐私的各种途径,并设计了一个评估工具 Pluto,能够自动评估应用的数据泄露程度.文献[3]分析了广告平台具体收集了哪些数据作为个性化广告的依据,包括兴趣爱好(运动、电影等)和人口统计信息(性别、年龄和种族等).

在广告主方面,其发布的广告内容也可能是恶意的[7−10].例如,文献[7]研究发现:由于广告库对广告内容的隔离不充分,使得广告主能够在展示的广告内容中加入一些可执行脚本.这些脚本可以绕过广告库的权限隔离,窃取用户隐私.另外,访问外部存储功能是广告内容一般具有的权利,即使不能读取文件的内容,广告主也可以根据缓存文件的名称推测用户的隐私.例如,根据药物名称推测用户的患病情况,根据交友图片推测用户的社交情况等.这种攻击依赖于宿主应用在使用过程中以文件的形式保存一些浏览记录或者缓存,攻击范围较窄.而且,攻击者需要支付一定的费用来发布恶意的广告内容,攻击成本较高.

在移动应用方面,除了直接的恶意软件之外,应用开发者能够通过技术手段欺骗广告平台,以获取非法利润[3,11,12].例如,文献[11]研究表明:恶意的移动应用可能通过大量设置模拟器环境,自动执行和触发自身展示的

广告来欺骗广告平台.另外,移动应用也可以根据定向广告的内容来间接地推测用户隐私.例如,文献[3]研究表明,个性化广告所关联的用户特征可能会泄露用户隐私.恶意的移动应用通过分析定向广告展示的具体页面,可以知道用户是否购买了某种商品,以及具有某种兴趣爱好等.

综上所述,现有的攻击方案从不同的角度给移动广告行业带来安全威胁,造成的危害也不容忽视.但是这些攻击方法针对不同的终端和宿主应用都采用相同的攻击载荷,方法过于"通用".事实上,不可能存在一种攻击方法能够适用于所有的场景,一旦宿主应用没有相应权限或终端不存在相应的漏洞,那么贸然发动攻击必然会引起用户和安全检测软件的警觉.因此,定制化的攻击载荷更适用于现有的 Android 防护机制.

1.2 威胁模型

为了方便开发者设计更灵活的功能和更好的兼容性,Android 提供了一种称为混合应用框架(hybrid APP framework)的开发模式,例如 PhoneGap,MoSync 和 appMobi 等.混合应用框架主要包括两部分:本地层和 Web 层[16].其中,本地层的最核心部分为操作系统,负责管理设备的各种资源(例如摄像头、通讯录、位置等);Web 层以浏览器的形式(WebView)嵌入到框架中,能够与本地层代码进行交互.为了保证本地层和 Web 层交互的安全性,混合应用框架提供了一些防护机制.例如,在 Web 层,混合应用的访问控制主要通过同源策略(same origin policy,简称 SOP)进行管理,能够阻止读写本地资源的非同"源"JavaScript 代码执行;在本地层,混合应用的安全性主要依赖于操作系统的访问控制策略,包括身份认证和权限机制等.

移动应用嵌入广告同样采用的是混合应用框架,主要包括两个部件:AdSDK 和 WebView,如图 2 所示.其中,AdSDK 是广告平台提供给开发者用来处理广告请求和展示广告的依赖库,实现了广告相关的功能接口,移动应用在启动之后,AdSDK 调用相关接口就能发送广告请求,并且解析广告平台服务器发送来的广告信息;WebView 可以理解为一个轻量级的浏览器,实现了一些基本功能,例如 HTML 页面的解析、JavaScript 运行环境和图形的渲染等.与 PC 端浏览器访问本地资源时弹出提示对话框不同,WebView 作为一种内嵌式浏览器,能够自动访问本地设备的资源(例如摄像头、日历、通讯录、短信和图片等),很难做到每次访问都请求用户的许可.因此,WebView 的安全性非常依赖于 Android 系统自身的防御措施.

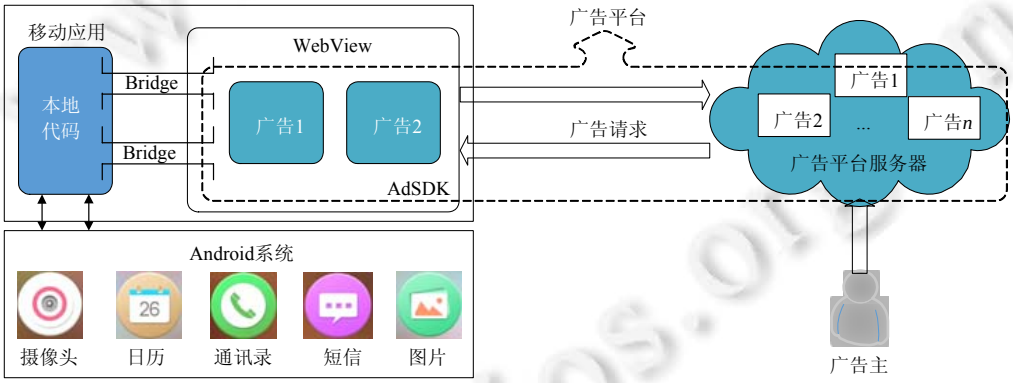


Fig.2 Hierarchical structure of mobile advertising ecosystem

图 2 移动广告生态系统层次结构

一般情况下,应用开发者不希望广告代码在本地执行,只允许页面展示.但是,很少有应用开发者能够对混合应用框架有深入的了解,而是将安全性寄希望于框架本身.混合应用框架的安全性依赖于一种潜在的信任关系.例如,用户授予权限给应用,但是不会授予权限给应用中的广告.应用的开发者信任广告库,将广告库的代码嵌入到自己的"源"中,这样,广告库就能创建 iframe 并展示广告.此时,同源策略能够将广告内容隔离在 iframe 中,以阻止其访问其他资源.但是,为了方便同源的 Web 内容执行 JavaScript 代码,混合应用框架提供了特殊的通道,称为 Bridge.该通道虽然能够满足开发者的需要,但是却存在一些漏洞,使得外源的 Web 内容绕过同源策略的检测.更严重的是,应用开发者在实现 Bridge 机制时,经常没有很好地执行同源策略,从而导致漏洞的产生.

本文的攻击方案在广告网络流量中注入恶意广告内容,发起中间人攻击(man-in-the-middle attack,简称MITM).攻击者位于合法主机的通信路径中间,通过捕获、修改和转发双方之间的数据包来达到攻击目的.本文利用混合应用框架的隔离漏洞来访问本地资源,攻击目标包括窃取隐私(通讯录、位置等)、网络欺诈(以虚假的身份骗取个人账号和密码等)、获取利润(向指定号码发送短信)、拒绝服务(导致手机不停震动或鸣叫)和破坏系统(删除联系人、短信等).本文方案假设广告平台、广告主和移动应用都是可信的,不会主动窃取用户隐私.本文的攻击者拥有一个远程服务器和恶意域名,并能够在控制的 Web 站点上执行恶意 JavaScript.

2 系统设计

2.1 系统框架

本文提出一种基于宿主权限的移动广告漏洞攻击方法,能够自动分析宿主应用的权限和广告服务商的隔离漏洞,并生成相应的攻击载荷.通过将定制后的攻击载荷植入到广告内容中,能够绕过安全软件的检测,攻击手段更加隐蔽.而且,根据宿主权限和特定漏洞发起的攻击能够避免失败时系统的异常报警,不易被用户察觉.本文攻击方案的系统框架如图 3 所示,主要包括以下 4 个功能模块.

(1) 流量分析.该模块利用流量识别算法从普通网络流量中提取出广告流量,并且利用机器学习分类算法从广告流量中识别出宿主应用的标识和设备相关信息,包括名称、版本号等.

(2) 权限分析.该模块根据宿主应用的标识,利用爬虫工具从官方应用市场(Google play store)中获取其权限列表,确定攻击者在当前环境下的能力上限.

(3) 漏洞分析.该模块深入分析现有移动应用对广告库的隔离措施,并根据 Android 系统版本和设备相关信息确定待攻击广告库 AdSDK 的 WebView 安全漏洞,以绕过宿主应用对广告内容的隔离.

(4) 载荷生成.该模块根据宿主应用的权限和隔离措施的漏洞,提出一种攻击者能力描述语言,能够自动生成可行的攻击载荷并植入到广告内容中返回客户端.

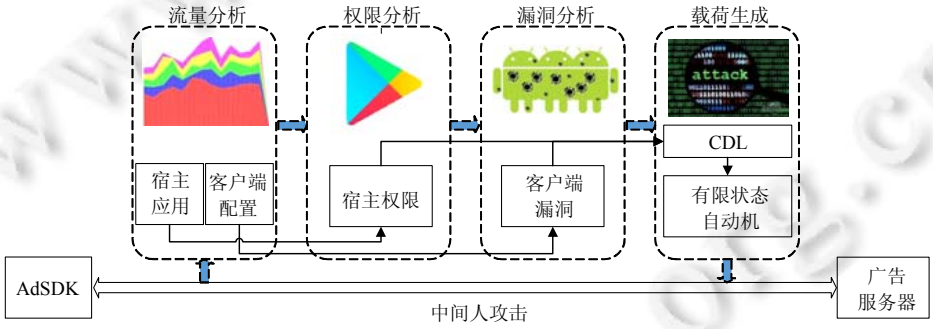


Fig.3 System architecture

图 3 系统结构图

2.2 流量分析

广告平台为了确定应用的身份,会在获取广告请求时将宿主应用的标识信息一起发送,这样便于广告费用的支付.广告库 AdSDK 在应用启动之后会立即向广告服务器发起请求,而应用本身也会产生一些网络流量.因此,本文首先结合深度包检测和深度流检测技术将广告流量和应用流量区分开,然后再利用半监督学习方法识别出宿主应用的标识和客户端相关信息.

网络流量以 flow 表示,每个 flow 由一系列的特征组成,每个特征从 1 个或多个数据包中得到.本文利用 NetMate[17]工具来分析数据包,生成相应的 flow 并计算特征值.网络流量 flow 用一个五元组(源地址,源端口,目的地址,目的端口,协议)表示,代表了两个主机之间的数据包交换.另外,对每个 flow,计算出一些统计信息作为特征,例如周期、字节、数据包传输时延均值和数据包大小均值等.简单来说,深度包检测主要关注流量的固有特

征,深度流检测主要关注流量的一些统计特征.最终,本文共提取出 15 个特征,见表 2.

Table 2 Feature selection and description

表 2 特征选取及说明

技术	特征	说明
深度包检测	源 IP、目的 IP	客户端和服务器各自的 IP 地址
	源端口号、目的端口号	客户端和服务器各自的端口号
	协议类型	宿主应用网络通信的协议类型
	应用层负载内容	数据包负载内容签名
深度流检测	包长序列	流中特定位置包的方向,包括上行或下行,以及包长的范围
	包长集合	流中连续或不连续的包长数字集合
	包长范围	包长的连续范围
	包长平均值	流中某个方向连续或不连续的包长的平均值
	简单包长求和	流中连续的几个数据包的包长之和
	轮次包长求和	与简单包长类似,只不过循环出现
	包长复现	指定长度、条件、位置和方向的包长复现
	包数统计	流中连续或不连续的方向上行或下行的包数
	报文收发比	接收或发送报文中流量较大一方与较小一方的比值

为了从广告流量中识别出宿主应用标识和客户端系统版本,本文用 $X=\{X_1,\ldots,X_N\}$表示网络流量 flow 集合,X_i 由特征向量表示.其中,$X_i=\{X_{ij}|1\leqslant j\leqslant m\}$,$m$ 表示特征的数量,X_{ij} 表示第 i 个 flow 的第 j 个特征取值.令 $Y=\{Y_1,\ldots,Y_q\}$表示宿主应用的标识,$S=\{S_1,\ldots,S_p\}$表示客户端系统版本.其中,q 表示嵌入广告的宿主应用数量,p 表示当前 Android 系统的所有版本数量.

目前最直接的分类方法是监督学习算法,能够根据 N 个训练数据元组(X_i,Y_i)训练得到一个映射关系 $f(X)\rightarrow Y$.对于一个未知类别的样本,代入到映射 $f(X)$中,就可以得到所属类别.但是对于本文的流量分析和识别方法,单纯的监督学习算法面临两个挑战.

- 有标签流量样本稀少并且难以获得.在已知标签过少时,监督学习训练得到的分类算法可能无法覆盖到所有的特征,对于未知样本可能错误分类.
- 并不是所有应用的流量特征都是固定的.随着版本的升级,应用产生的网络流量可能会发生变化,传统的有监督学习算法只能将已知的样本集分为 q 类,难以扩展到其他新类别.

为了解决监督学习算法存在的问题,本文结合监督学习和无监督学习提高识别效率和准确率,称为半监督学习[18].半监督学习主要考虑如何利用少量的标注样本和大量的未标注样本进行训练和分类的问题.具体来说,本文的流量分析方法分为两步:聚类和匹配.聚类的目的是将有标注样本和无标注样本按照相似度聚集在一起形成多个数据集.这样,当两个样例位于同一聚类簇时,它们在很大的概率下有相同的类标签.匹配的目的是将这些聚集在一起的数据集对应到 q 个类别.接下来,本文详细介绍流量分析模块的细节.

2.2.1 聚 类

聚类分析是典型的无监督学习算法,能够从所有可用的网络数据流量中得到若干个聚类簇.一个聚类簇之内的样本相似度较高,不同的聚类簇之间相似度较低.通过计算数据流之间的相似度,聚类分析能够识别出应用和系统标识.对于两个特征向量 x_i 和 x_j,两者的相似度通过计算距离 $d(x_i,x_j)$得到.本文利用欧几里得距离作为两个数据流量之间的相似度,计算公式如下:

$$d(x_i,x_j)=\left[\sum_{k=1}^{m}(x_{ik}-x_{jk})^2\right]^{1/2}.$$

在机器学习领域,有非常多的聚类分析算法,例如 K-Means、DBSCAN 和 EM 等.本文利用 K-Means 算法[19]进行实验.与其他聚类算法相比,K-Means 算法简单且容易实现,只需要非常短(分钟级)的学习时间就能达到较高的准确率.在聚类时,K-Means 算法随机选取 k 个流量作为初始聚类的中心,初始地代表一个簇.对数据集中的剩余样本,每次迭代计算其余各个簇中心的距离,将相近的样本重新赋给最近的簇.当计算完所有流量样本之后,一次迭代运算完成,新的聚类中心形成.如果在一次迭代前后,聚类中心没有发生变化,那么聚类算法就已经收敛.该算法的时间复杂度为 $O(lKn)$,其中,l 表示迭代的次数,n 表示训练数据集中样本的个数.在本文实验过程

中,*K*-Means 算法能够在几次迭代之后收敛.

2.2.2 匹 配

K-Means 算法的输出是一系列的聚类簇,用各自的中心γ_k表示.给定一个数据流特征向量x,通过找到其最接近的中心,能够找到其所属的簇:

$$C_k = \arg\min_d d(x,\gamma_k),$$

其中,$d(x,\gamma_k)$表示当前样本与中心γ_k的距离.

接下来,需要将每个聚类簇对应到具体的宿主应用和系统版本.以宿主应用的识别为例,本文计算一个概率公式$P(Y=y_j|C_k)$,其中$j=1,\ldots,q$表示应用的个数,$k=1,\ldots,K$表示聚类簇的个数.在数据集中,有L个样本已知应用标签,用(x_i,y_i),$i=1,\ldots,L$表示.$P(Y=y_j|C_k)$通过计算最大似然估计n_{jk}/n_k,其中,n_{jk}表示聚类簇k中数据类别j的数据流个数,n_k表示聚类簇k中所有的样本个数.

最终,对于一个特征向量x,其类别由一个后验决策函数确定:

$$y = \arg\max_{y_1,\ldots,y_q}(P(y_j \mid C_k)),$$

其中,C_k表示最接近x的聚类簇,即宿主应用的标识.特殊情况下,如果x没有找到一个聚类簇,那么本文将其标记为“未知”应用,需要攻击者人工进行确认.

2.3 权限分析

从广告流量中提取出宿主应用的标识之后,下一步需要得到其申请的权限列表.直观上,分析 Android 应用的权限列表可以通过对 APK 文件进行分析,从配置文件 AndroidManifest.xml 文件中获得.但是本文攻击方案是在网络中实时运行,而不是在终端上离线分析.因此,攻击者无法直接获得当前宿主应用的 APK 文件.为了解决这个问题,本文设计了一种从应用市场中分析元数据以得到权限列表的方法.观察发现:很多应用市场在推荐用户安装时,除了给出应用的功能介绍以外,还会给出该应用的权限情况,以告知用户可能存在的安全风险.因此,通过对宿主应用元数据的爬取,攻击者在不需要静态分析的情况下,也能快速得到该应用的权限列表.

出于对安全性和自身利益的考虑,Google 官方市场对应用的访问和下载设置了一些防护措施,使得简单的爬虫程序无法直接得到应用的元数据.为此,本文利用 PlayDrone[20]工具设计了一个定制化的爬虫工具,能够成功地绕过官方市场对主机下载应用的限制.截止到 2017 年 5 月 1 日,共获得 1 402 894 个官方 APK 文件的元数据,包括 apk 的下载地址、下载次数、开发者名称、权限列表等.这些元数据以 json 文件的形式存在,能够快速地进行查找.另外,相同宿主应用的不同版本之间也可能申请不同的权限.因此,本文在分析当前宿主应用的权限列表时,将版本信息也考虑在内,与应用名称一起与元数据中的内容进行匹配.

2.4 漏洞分析

如第 1.2 节所述,移动应用嵌入广告的方式是一种混合应用模式,这种模式存在一些漏洞,使得攻击者能够绕过广告库隔离措施的限制.目前,已经有一些研究人员在不同的 Android 版本上发现了大量可行的攻击途径[2−12].通过对这些攻击途径的分析和总结,在确定了客户端操作系统和应用程序信息之后,本文攻击者能够自动选择可行的攻击方法.具体来说,本文主要关注 7 类 WebView 相关漏洞,包括 WebView 任意代码执行漏洞、WebView 域控制不严格漏洞、WebView File 域同源策略绕过漏洞、Web 层同源策略解析漏洞、Web 层和本地层语义误差漏洞、外部存储文件推测漏洞和其他侧信道攻击.限于篇幅,具体的漏洞细节不在文中详细介绍.下面以 WebView 任意代码执行漏洞为例作简单说明.

Android 系统为了方便应用中 Java 代码和网页中的 JavaScript 脚本交互,在 WebView 控件中实现了 addJavascriptInterface 接口.通过该接口,网页中的 JS 脚本能够调用应用中的 Java 代码,而 Java 对象继承关系会导致很多公开的函数以及 getClass 函数都可以在 JS 中被访问.结合 Java 的反射机制,攻击者还可以获得系统类的函数,进而可以进行任意代码执行.以 CVE-2012-6636 为例,攻击者首先在 WebView 中添加 JavaScript 对象,并且申请一些权限,比如想要获取 SD 卡上面的信息就需要 WRITE_EXTERNAL_STORAGE 权限.然后在 JS 中

遍历 Window 对象,找到存在 getClass 方法的对象,通过反射机制,得到 Runtime 对象.接下来可以调用静态方法来执行一些命令,比如访问文件的命令.最后,从执行命令后返回的输入流中得到字符串,比如文件名的信息等.虽然该漏洞在 2013 年 8 月被批露,但是截至目前,仍有很多应用存在此漏洞.很多应用开发团队因为缺乏安全意识,仍然在应用中随心所欲地使用 addJavascriptInterface 接口,从而能够被攻击者利用.

2.5 载荷生成

Android 系统的安全特性是通过权限机制对特定进程的特定操作进行限制,应用程序在默认的情况下不可以执行任何给其他应用程序、系统和用户带来负面影响的操作.一个应用程序的进程就是一个安全沙盒,除非显式地声明了相应权限,否则它不能获取沙盒不具备的额外能力.广告内容存在于宿主应用的进程空间中,如果宿主应用缺少必要的权限,那么广告库和广告内容也不能获得所期望的功能.在生成攻击载荷时,一旦贸然调用敏感的资源或者访问受限的数据,那么一定会触发 Android 防护系统的报警,从而引起用户的察觉.

事实上,宿主应用的权限列表代表了移动广告攻击者的能力上限,攻击者应该根据宿主应用的能力制定相应的攻击策略.例如,如果宿主应用申请了 ACCESS_FINE_LOCATION 权限,攻击者就可以利用一些隔离漏洞访问用户的位置.如果宿主应用申请了 READ_SMS 权限,攻击者就可以获取用户的短信息.相反,如果宿主应用没有申请这两个权限,攻击者在访问位置和短信时就会引起异常.因此,本文基于宿主应用的权限列表,提出了一种自动生成载荷的定制化攻击方法,能够最大程度地避免 Android 系统察觉.

在下一节中,本文将会提供自动生成攻击载荷的更多设计和实现细节.

3 基于宿主权限的攻击载荷自动生成方法

本节详细介绍基于宿主权限的攻击载荷自动生成方法,能够根据不同应用的权限申请、不同客户端系统的 WebView 安全漏洞自动选择最优的攻击技术并生成相应代码.攻击载荷的生成步骤如图 4 所示,共分为 3 个模块:(1) 能力描述语言生成;(2) 能力解析;(3) 载荷生成.能力描述语言生成模块负责用一种统一的格式表示前期收集和分析到的宿主能力,为载荷的生成提供数据支撑;能力解析模块负责对生成的描述语言进行分析,确定可用的攻击技术;载荷生成模块负责组装具体的攻击代码,并植入到广告内容中返回宿主应用.

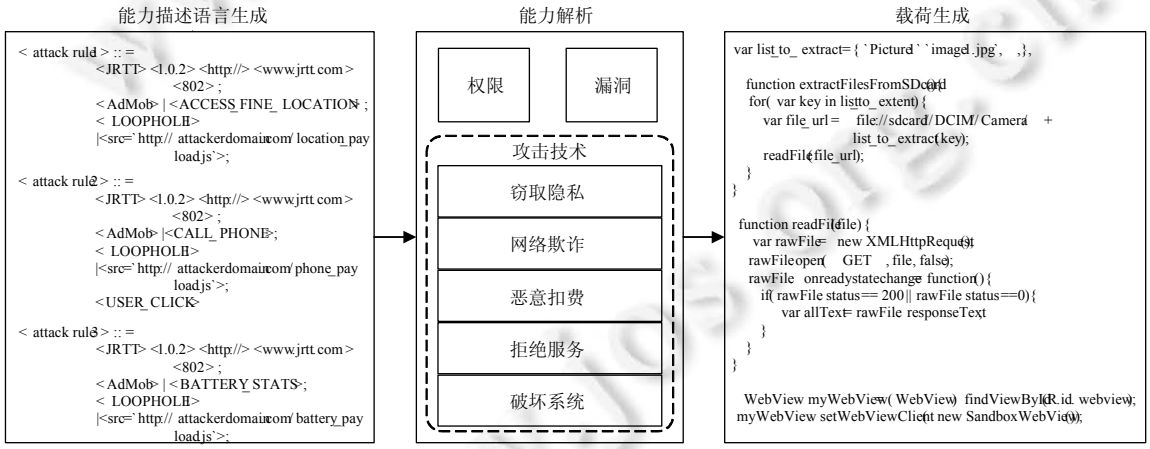


Fig.4 Stages of automatic attack code generation

图 4 自动生成攻击载荷过程

3.1 能力描述语言生成

为了便于计算机自动处理宿主应用和客户端系统的相关信息,本文提出一种能力描述语言 CDL(capability description language).CDL 能够用来标记权限、定义权限类型、描述 WebView 漏洞等,提供统一的方法来描述宿主应用的结构化数据.CDL 语言的定义如下.

定义 1(CDL 能力描述语言 capability description language). CDL 是一种数据交换格式,用来传输和存储宿主的能力信息.CDL 规定了一些特定的标签,能够用来描述各种软件的配置数据.CDL 的目标是实现细粒度的宿主能力描述,能够对攻击者可以利用的所有使用场景的所有访问通道进行控制.使用场景表示攻击者能够达到的攻击目标,对应于宿主应用的权限列表.访问通道表示攻击者能够利用的攻击途径,对应于客户端系统的漏洞信息.

本文利用巴科斯范式(Backus-naur form,简称 BNF)来描述 CDL 的语法,允许攻击者制定策略来明确能够从广告内容中获取的敏感资源.一个典型 CDL 的语法规则如下:

⟨attack rule⟩::=⟨target app⟩;⟨subject⟩|⟨permission⟩;⟨channel⟩;⟨related point⟩.

每一条攻击规则都对应于一个目标应用.⟨target app⟩标签表示广告库嵌入的宿主应用标识,包括应用名称、版本、域名、通信协议和端口.为了允许攻击者发起通用的攻击策略,CDL 允许采用正则表示式来描述各类标签信息,如下所示.

⟨target app⟩::=*|⟨name⟩⟨version⟩⟨protocol⟩⟨hostname⟩⟨post⟩,

其中,⟨protocol⟩::=⟨http://⟩|⟨https://⟩.

每个宿主应用中可能包含多个广告展示位,⟨subject⟩对应于每一个广告库的名称.每个广告库申请的权限列表不同,⟨permission⟩表示宿主应用需要申请的权限.在当前的攻击规则下,可能不需要权限,也可能是 1 个或者多个,如下所示.

⟨permission⟩::=⟨permission⟩|⟨permission list⟩|∅.

本文攻击者能够利用的攻击途径包括第 2.4 节中介绍的所有已知漏洞,用⟨channel⟩表示.为了使攻击者能够制定细粒度的攻击策略,每一条攻击途径都需要明确、具体的攻击漏洞和利用方法.特别地,对于访问控制策略审查严格的攻击途径,需要进一步明确需要申请的权限,如下所示.

⟨channel⟩::=⟨loophole⟩|⟨methods⟩|⟨permission list⟩.

另外,为了达到某些攻击规则,需要用户的一些操作支持,包括用户点击、文本输入等,用⟨related point⟩表示.这些相关信息使得攻击者能够了解该条攻击策略的先决条件,如下所示.

⟨related point⟩::=⟨user click⟩|⟨text input⟩|∅.

值得注意的是,对于任何一种表达语言,都需要在表达性和可用性之间寻找折中点.一方面,可用的表达语言应该是简单的,但是会影响到表达性;另一方面,复杂的语言表达能力很强,能够表示非常细粒度的攻击策略,但是会影响到可用性.作为一种攻击方法,本文的目标是在不被用户察觉的基础上执行一定的非法操作,因此,CDL 语言重点关注经过验证的攻击策略,以保证在宿主能力满足的前提下的攻击成功率.

3.2 能力解析

该模块负责对 CDL 文件进行解析,以制定相应的攻击策略.CDL 文件的解析方法非常直观,通过对每一条攻击策略的分析,在预先定义好的攻击数据库中选择匹配的攻击技术.这种定制化的攻击方法能够在不同的应用场景下执行不同的攻击行为,相对于其他攻击形式更为高级和先进.与高级持续性威胁(advanced persistent threat,简称 APT)技术类似,本文提出的攻击方法在发动攻击之前同样需要对宿主应用的能力进行精确的收集和解析.目前,针对移动设备的攻击技术有很多,包括恶意软件、数据泄露、僵尸网络和设备监听等.受到 Android 系统 WebView 隔离措施的限制,并不是所有的攻击技术都能通过广告内容的可执行脚本发起攻击.因此,需要根据实际情况选择具体可用的攻击技术.

下面简单介绍针对移动广告可用的攻击技术,相应的能力要求见表 3.

1) 窃取隐私

敏感数据泄露是移动设备最大的风险之一,对于攻击者来说具有极大的吸引力和价值.随着移动设备越来越私有化,移动用户倾向于在手机上存储各种类型的数据,包括照片、联系人、短信和聊天记录等.本文关注的隐私权限包括:获取设备信息、读取位置信息、打开 WiFi、访问联系人、发送短信、读取短信记录、拨打电话、读取通话记录、监听手机通话、使用话筒录音、打开数据开关、读取浏览记录等共 12 项权限.根据攻击

者的指令,可以将搜集的用户数据上传到指定的服务器上.

Table 3 Summary of five attack technologies

表 3 5 种攻击技术总结

序号	攻击技术	基本原理	能力要求	攻击难度	攻击效果
1	窃取隐私	访问本地资源,获取用户和设备的各种信息	1) 宿主应用具有读取各种用户数据的权限 2) WebView 中的广告内容能够与宿主应用的本地代码通信	易	好
2	网络欺诈	获取用户身份信息后冒名顶替,欺骗用户上当受骗	1) 攻击者获取到用户的各种身份信息,包括账户、用户名等 2) 利用社会工程学或网络钓鱼方法欺骗安全意识较差用户	难	一般
3	恶意扣费	利用智能手机的几种通讯功能带来高额的话费开销	1) 宿主应用具有发送短信或者拨打电话权限; 2) 攻击者在电信运营商注册有增值业务	一般	一般
4	拒绝服务	阻止合法用户正常使用设备和各种应用提供的服务	1) 宿主应用能够访问网络、加速计、陀螺仪、话筒等传感器; 2) 系统打开相关的硬件功能	易	差
5	破坏系统	给用户设备造成直接的硬件或者软件破坏	1) 宿主应用申请相关权限 2) 攻击者判断设备空闲	难	差

2) 网络欺诈

网络欺诈是指采用虚构事实或者冒名顶替的方法,通过互联网骗取用户信任并且侵吞公私财物的行为.本文关注的网络欺诈行为主要是获取用户的各类身份信息,包括 QQ 用户名、联系人、年龄、性别、收入等.基于这些身份信息,攻击者能够进一步获得用户的社交人脉网络.然后利用网络钓鱼或者社会工程学方法,向用户本身或者好友发送虚假信息,欺骗用户上当.

3) 恶意扣费

攻击者获取利润的直接方法就是骗取用户开通增值业务,通过向指定的业务号码自动发送短信的形式能够进行恶意扣费.在发送短信的过程中,攻击者还会拦截运营商短信,能够在用户毫不知情的状态下绕过二次确认过程.除了外发短信的形式以外,攻击者还能在后台控制手机自动拨打指定的业务号码,以收取高额的费用.另外,攻击者还可以通过自动刷新联网应用的点击量来不断消耗用户的上网流量,同样会造成相当程度的资费损失.

4) 拒绝服务

拒绝服务攻击是一种常见的黑客攻击手段,能够让目标机器停止提供服务甚至死机.在传统 PC 平台,攻击者采用的手段主要包括两类:(1) 迫使服务器的缓冲区满,不接收新的网络连接;(2) 使用 IP 欺骗,迫使服务器把非法用户的连接复位.针对 Android 平台,本文采用的拒绝服务方法与 PC 有所不同,主要通过对设备本身传感器的干扰达到用户无法正常使用的目的,包括手机不停震动、话筒持续发出蜂鸣声、应用程序异常退出等.

5) 破坏系统

前面几种技术都属于非破坏类的攻击手段,在达到攻击目的之后不会损坏设备的硬件或者数据.而破坏性攻击是以破坏目标系统的数据为目的,并不盗窃用户的隐私信息.在本文的攻击场景下,由于受到 Android 权限机制的控制,有些设备可能无法达到上述 4 个攻击效果.在这种情况下,即使攻击者不能直接获取用户的各种资源,也可以选择对系统进行直接破坏,例如删除用户的短信和通讯录、卸载已安装应用等.这种攻击方法容易被用户察觉,因此只是作为一种辅助手段.

3.3 载荷生成

通过对宿主权限和客户端漏洞的分析,载荷生成模块能够自动选择可用的攻击技术,并合成代码.该模块最大的挑战在于如何根据 CDL 文件的能力描述,从上述攻击技术中选择最合适的方法.解决这个问题的最简单方法是尝试所有的代码组合,并判断每种组合的攻击效果.显然,这种方法对于实时性要求较高的中间人攻击非常不适用,很容易引起 Android 防护系统的察觉.本文利用有限状态自动机[21]实现了一种广度优先搜索(breadth-

first search,简称 BFS)方法,以设置不同攻击技术之间的转换和触发顺序.该转移函数能够确定在当前的能力状态下哪种攻击技术适用,而且如何与已植入的攻击代码融合.例如,CDL 文件中有读取短信的权限时,首先在当前状态下植入获取隐私的攻击载荷,然后再发送扣费短信以进一步获取利润.给定能力状态之后,只有满足转换条件的技术才会被加入到攻击载荷中.

定义 2(有限状态自动机 finite state automata(FSA)). 有限状态自动机是一个五元组(Q,Σ,δ,q_0,F),其中,Q 是一个有穷集合,称为状态集;Σ是一个有穷集合,称为字母表;$\delta: Q\times\Sigma\to Q$ 是转移函数;$q_0\in Q$ 是起始状态;$F\in Q$ 是接收状态集.

攻击载荷生成过程可以看成是一个软件黑箱,它可以与攻击者进行交互.攻击者通过输入 CDL 文件调用载荷生成过程,CDL 文件中的宿主权限和广告库漏洞激发内部逻辑判断,然后输出可用的攻击代码.本文将 CDL 文件中的权限和漏洞按照风险高低划分成几个子文件,攻击者由高到低依次将每个子文件作为载荷生成过程的输入.因此,可以用有限状态自动机来描述攻击载荷生成过程,即攻击者的每次交互可被看成有限状态自动机的状态转换.

本文用一个六元祖(Q,I,O,δ,q_0,F)来表示攻击载荷的自动生成过程,其中,

- $Q=\{q_0,\ldots,q_n,q_a,q_b,q_c,q_d,q_e\}$表示状态集,$q_0,\ldots,q_n$ 表示权限和漏洞列表,$q_a,\ldots,q_e$ 表示 5 类攻击技术;
- I 是输入 CDL 文件的有穷集合;
- O 是输出攻击代码的有穷集合,$I\times O$ 是 FSA 的字母表;
- $\delta: Q\times I\times O\to Q$ 是转移函数,即给定一个能力,根据$\langle I,O\rangle$,FSA 能够转移到另一个状态;
- $q_0\in Q$ 是起始状态;
- $F=\{q_a,q_b,q_c,q_d,q_e\}$是接收状态集,即攻击者能够与 FSA 结束交互的状态集(q_a=窃取隐私,q_b=网络欺诈,q_c=恶意扣费,q_d=拒绝服务,q_e=破坏系统).

攻击载荷自动生成过程的形式化模型可以定义为 $FSA=(Q,I,O,\delta,q_0,F)$.

输入集合 $I=\{READ\_SMS,ACCESS\_FINE\_LOCATION,\ldots,loophole1,loophole2,\ldots,loophole7\}$.

输出代码集合 $O=\{getLocation(),sendMessage(),\ldots,delPhoneBook()\}$.

字母表:

$I\times O=\{READ\_SMS/sendMessage(),ACCESS\_FINE\_LOCATION/getLocation(),\ldots,loophole1/delPhoneBook()\}$.

转移函数δ为

$$q_0\times ACCESS\_FINE\_LOCATION/getLocation()\to q_a,$$
$$q_a\times READ\_SMS/sendMessage()\to q_c,$$
$$\ldots$$
$$q_0\times loophole1/delPhoneBook()\to q_e.$$

一个有限状态自动机可以表示成一个状态转换图,如图 5 所示.图中节点表示不同的能力状态,节点之间的连线表示当前状态下选择哪种攻击技术.首先,攻击载荷的初始状态(q_0)为空,根据权限列表尝试加入对应的隐私窃取(q_a)方法.本系统将隐私窃取作为基础的攻击方法,主要是因为结合先进的数据挖掘技术,任何获取到的信息都有可能泄露用户的隐私,从而给攻击者带来价值.然后,根据窃取到的隐私($\{q_1,\ldots,q_n\}$)确定是否进行网络欺诈(q_b)或者获取利润(q_c).如果在隐私窃取状态下能够得到用户的身份信息(例如用户名、通讯录等),就可以在攻击载荷中加入网络钓鱼类(q_b)的代码.如果能够读取并控制设备的短信服务,就可以向指定业务发送扣费短信.相应地,如果能够控制电话服务,就可以直接拨打扣费电话.最后,根据前两步的攻击载荷组合情况,确定是否需要进一步植入拒绝服务(q_d)和破坏系统(q_e)攻击.这两种攻击方法很容易引起用户的察觉,因此,只有在前 3 种攻击手段无法达到效果时才选择植入.

为了降低代码开发时编写、修改和调试的工作代价,本文利用 Metasploit[22]框架中的模块化攻击载荷植入代码.Metasploit 提供了大部分流行操作系统上功能丰富多样的攻击载荷模块,包括 Android,Linux,Unix,Windows 和 Mac OS X 等.这样,本文就可以在确定宿主应用的权限和广告网络的漏洞之后,从很多适用的攻击

载荷中选取最可靠的模块进行灵活组装.并且在渗透攻击之后,还可以选择控制会话类型,极大地提高了攻击的成功率.

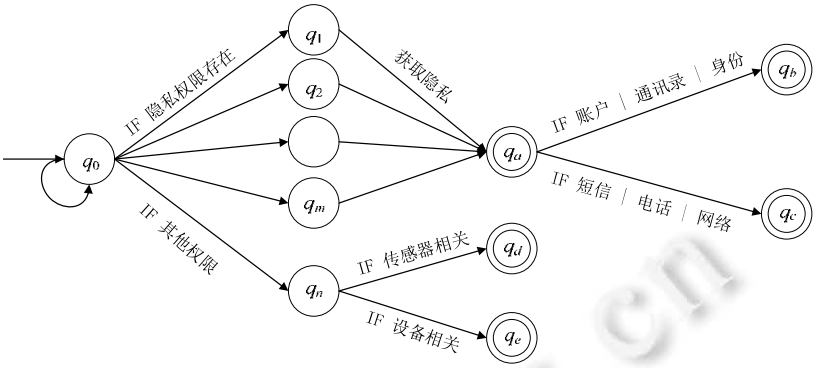


Fig.5 Example of status transformation graph

图 5 状态转移图实例

3.4 攻击实现

在移动广告中间人攻击的具体实施中,本文利用 ARP 欺骗获取宿主应用与广告服务商之间的通信数据流的控制权.具体的攻击步骤如下.

- 第 1 步,启用中间人攻击主机,通过 ARP 欺骗方法,使移动应用在获取广告内容时认为广告服务商 IP 对应的 MAC 地址是中间人主机的 MAC 地址;同样,使广告服务商认为移动客户端 IP 对应的 MAC 地址是中间攻击人主机的 MAC 地址.该步骤利用 Dsniff 网络嗅探工具包实现,可以自动分析端口上收到的某些协议的数据包.
- 第 2 步,客户端运行宿主应用,后台自动发送广告请求到广告平台服务器,但实际上连接到的是中间人攻击主机.该步骤利用 *K*-Means 聚类分析算法从广告流量中识别出宿主应用的标识,并且分析出客户端的配置信息.注意:该步骤中间人攻击主机不会对广告请求进行修改,而是按照原有格式转发给广告平台服务器.
- 第 3 步,攻击者根据第 2 步分析出的宿主应用标识,向官方应用市场爬取该应用的权限列表.该步骤利用 Scrap 实现,它是 Python 开发的一个快速、高层次的 Web 抓取框架,能够抓取 Web 站点并从页面中提取结构化的数据.
- 第 4 步,攻击者根据第 3 步分析出的客户端配置信息检测设备漏洞,重点关注当前操作系统版本的 WebView 相关漏洞.该步骤利用 dSploit 工具对移动设备进行漏洞诊断,包括端口扫描、信息获取等.
- 第 5 步,根据宿主应用的权限以及客户端的漏洞,生成对应的 CDL 能力描述文件.
- 第 6 步,对 CDL 文件进行解析,利用 Metasploit 框架自动生成攻击载荷.
- 第 7 步,广告平台返回广告内容到客户端,但实际上连接的是中间人攻击主机.攻击者将第 6 步生成的攻击载荷植入到广告内容中,并返回客户端.

4 实验评估

4.1 实验环境

本文实验的硬件和网络环境如图 6 所示.其中,中间人攻击主机实施攻击和数据包的存储转发,基本设备包括多个大功率天线、一台配备有 BackTrack 操作系统的电脑,同时,还额外需要一块支持数据包注入的无线网卡.本文的客户端共包括 3 个设备:XiaoMi 5S,Huawei Mate8 和 Samsung Galaxy S6,它们各自搭载的 Android 版本为 6,5.5 和 5.0.2.中间人攻击主机运行在 ThinkPad T470 笔记本上,搭载 BackTrack5 操作系统,其中预装了

Metasploit 工具箱.

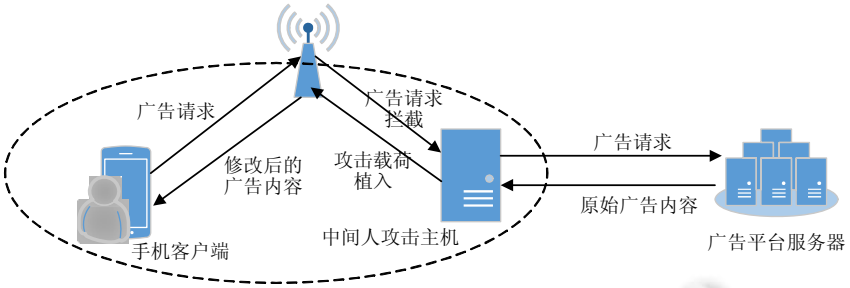


Fig.6 Hardware and network environment

图 6 硬件和网络环境

在数据集方面,本文利用 PlayDrone[20]从 Google Play Store 中下载了 17 245 个应用.由于并非所有的应用都含有广告,因此需要将它们区分开.为此,本文将这些应用在智能手机上依次运行,并捕获产生的网络流量.显而易见,依靠手工执行每个应用是不现实的,会消耗大量的人力成本和时间成本.所以,我们利用 Android 平台提供的自动化分析工具 MonkeyRunner[23]对每个应用进行测试,动态执行每个应用的每个页面.在应用的运行过程中如果嵌入有广告,那么必然会存在 AdSDK 与远程服务器的通信.通过在设备上运行网络嗅探工具或者设置代理,对捕获到的流量进行关键词匹配.如果查找到了广告服务商特定的关键词(例如 AdMob),那么相应的移动应用就被认为存在广告.最终,本文共得到 5 462 个应用含有广告内容.

4.2 实验结果

4.2.1 权限分析

本文对 5 462 个应用的权限申请情况进行统计,并给出了这些含有广告的应用中经常申请的 20 个权限,如图 7 所示.从图中可以看出,INTERNET、ACCESS_NETWORK_STATE 等网络相关权限申请数量非常多,超过 5 000 个应用都申请了这两个权限.另外,一些隐私相关的权限也经常出现,例如 ACCESS_FINE_LOCATION (2924),CALL_PHONE(1488),READ_CONTACTS(1464)和 SEND_SMS(866)等.这些权限可能是应用本身的需要,也有可能是广告服务商为了定制化广告而申请的.例如,为了丰富广告的展现形式,现代广告一般含有大量多媒体内容(视频、图片、语音等).由于这些多媒体内容的传输需要消耗网络带宽,所以大约有 3 682 个应用的 AdSDK 普遍通过申请 WRITE_EXTERNAL_STORAGE 权限将这些文件缓存在本地存储中.宿主应用申请的这些敏感权限无疑给非法人员利用广告内容发起攻击提供了可能.

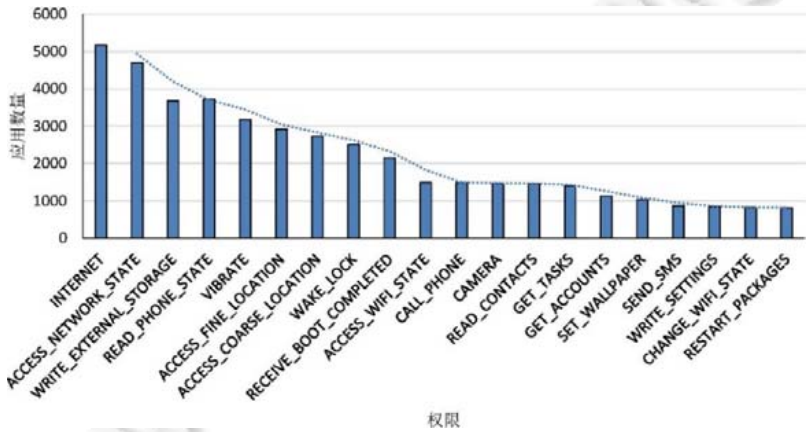


Fig.7 Distribution of permission requested by mobile apps with ad

图 7 含有广告的移动应用的权限申请分布

4.2.2 漏洞分析

本文对目前市场上占有率较高的几个 Android 版本的漏洞进行统计,结果见表 4.其中,漏洞 1~漏洞 7 分别代表本文在第 2.4 节列举出的 7 种 WebView 漏洞.从表中可以看出,Android 4.1 系统的漏洞最多,共发现 5 个已知漏洞.最新系统 Android 7.0 的漏洞最少,目前只有利用侧信道漏洞才能发起攻击.这说明,随着 Android 系统的不断升级,很多已知漏洞在新版系统中被成功地修复,也表明 Google 公司对 WebView 相关漏洞的高度重视.表 4 的最后一列也给出了每个版本的市场占有率情况[24],排名第一的是 Android Kitkat(6.0).该系统在 2017 年 5 月的占有率为 25.59%,共发现两个已知漏洞.对于最新系统 Android 7.0,在 2017 年 5 月的占有率仅为 6.33%.另外,有 3 个主流 Android 版本(4.4,5.0 和 5.1)的漏洞超过了 3 个,市场总占有率超过 55%.这主要是因为 Android 系统的碎片化问题造成的,最新系统很难及时发布到所有的 Android 手机上.这样,虽然新版的 Android 系统修复了一些已知漏洞,但是由于各大厂商更新速度不同,市场上仍然有大量的 Android 手机运行着旧版系统.随着 Android 市场整体占有率的提升,攻击者仍然可以从大量的旧版系统中成功地利用已知漏洞发起攻击.

Table 4 Distribution of loophole among different Android versions

表 4 不同 Android 版本的漏洞分布情况

Android 版本	漏洞 1	漏洞 2	漏洞 3	漏洞 4	漏洞 5	漏洞 6	漏洞 7	占有率(%)
7.0	–	–	–	–	–	–	√	6.33
6.0	–	–	–	–	–	√	√	25.59
5.1	–	–	–	√	–	√	√	22.52
5.0	–	–	√	√	√	√	√	10.44
4.4	√	–	√	√	√	√	√	22.53
4.1 以前	√	√	√	√	√	√	√	12.59

4.2.3 攻击效果

为了评估自动生成载荷的攻击效果和攻击范围,本文随机选择 1 000 个包含移动广告的应用进行实验.通过 Android 提供的 appt(Android asset packaging tool)工具,对这些应用的 APK 文件和相应的广告库 AdSDK 文件进行静态分析.其中,315 个应用嵌入的广告库中不包含可以执行 JavaScript 代码的 iframe,104 个应用通过 HTTPS 实现加密传输,剩余的 581 个应用通过 HTTP 协议向远端服务器动态获取广告内容.在这 581 个移动应用中,有一些应用处于休眠状态,没有应用开发者负责更新和维护,需要剔除出去.因此,本文进一步利用 MoneyRunner 动态执行工具对这些应用进行分析,确定哪些应用仍然可以发送和接收广告数据包.最终,本文共对 252 个应用进行实验.另外,虽然可以通过 SSLStrip 工具对 HTTPS 会话进行劫持,但是为了便于读者理解本文的攻击方案,暂不考虑通信加密的情况,而重点关注 HTTP 明文传输协议.而且,超过 58%的广告库传输使用的是 HTTP 协议,所以本文方案的攻击范围仍然很大.

图 8 表示宿主权限和广告库权限的分布情况.从图中可以看出:165(65%)个应用会申请超过 5 个权限,内部嵌入的 72(28%)个广告库 AdSDK 会申请超过 3 个权限.在宿主应用申请的权限列表中,有一些是自身需要,而有一些是为了广告库的需要.通过分析发现,广告库要求的权限一般会占到宿主应用所有权限的 20%.在这些广告库 AdSDK 申请的权限中,有很多都是与用户隐私相关的,例如 ACCESS_COARSE_LOCATION,READ_EXTERNAL_STORAGE 和 BROADCAST_SMS.为了广告的正常展示,很多宿主应用都没有遵循最小权限原则,会直接在自身的配置文件 AndroidManifest.xml 中加入广告库要求的权限.

通过人工对上述提取出的 252 个移动应用发起攻击,图 9 给出了 5 种攻击技术的实验结果.从图中可以看出:直接获取隐私的攻击效果最好,能够从 124(49%)个应用中窃取到用户的敏感数据;其次是拒绝服务和破坏系统这两种攻击方法,分别有 85(34%)和 63(25%)个应用被成功地攻击;然后是网络欺诈,能够从 31 个应用中获取到用户的个人信息;效果相对最差的是恶意扣费攻击方法,只有 5 个应用向预设的短信服务器发送了指定命令.这主要是因为该攻击方法会给用户带来直接损失,而且需要在电信运营商上注册相应业务.所以在本文的实验过程中没有将其作为主要攻击手段,只是用来验证方案的可行性.值得注意的是,在现实的攻击环境下,很多攻击者倾向于直接获取利益,所以这种攻击方法同样需要引起重视.

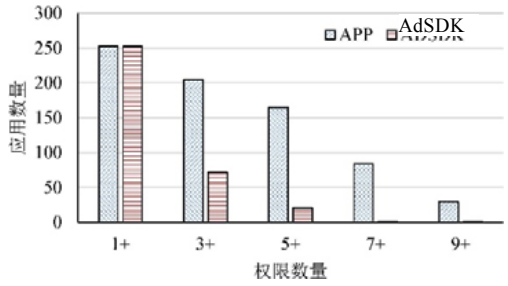


Fig.8 Distribution of permission request between app and AdSDK

图 8 宿主应用和广告库权限分布

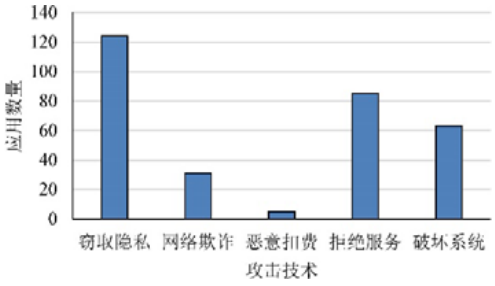


Fig.9 Attack effect

图 9 攻击效果

接下来,本文继续对攻击者能够获得的具体隐私数据进行分析,实验结果如图 10 所示.其中,获取设备信息和位置信息是攻击成功率最高的两大数据,分别能够从 103(41%)和 96(38%)个应用中获取.这与宿主应用的权限申请情况是相符的,即很多广告服务商为了提供定制化广告,需要获取这两类个性化信息,而这也恰恰被攻击者所利用.另外,对于访问联系人、获取通话记录和短信内容这样的敏感信息,本文方案也能获得不错的攻击效果,分别有 43 个、36 个和 33 个应用被攻击成功.而这主要归因于本文方案在发起攻击之前已经对宿主应用做了深入的能力分析,能够做到有的放矢,攻击也更有针对性.

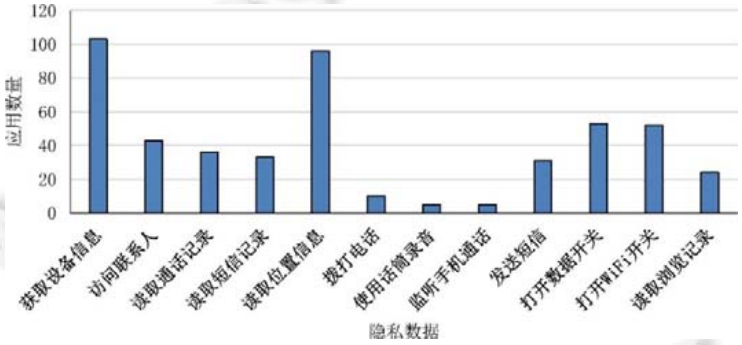


Fig.10 Attack result of different privacy data

图 10 不同隐私数据的攻击结果

4.3 攻击实例

为了更好地验证所生成攻击载荷的可行性,本节选择 4 个含有移动广告的 Android 应用进行详细介绍,包括 Angry Bird 2、盗墓笔记、nice 和 WiFi 万能密码.这 4 个应用的下载量都超过了 100 万次,各自嵌入了不同的 AdSDK,并且申请了不同的权限.本文将每个应用都分别安装到 3 个 Android 设备(Mi 5S,M8 和 S6)上,依次启动并激活攻击系统.具体的样本说明和攻击结果见表 5.

从表 5 中可以看出:在 3 个实验设备上,针对这 4 个应用的攻击均可以获得一些隐私数据.例如,Angry Bird 2 是一款免费的游戏应用,下载量超过 500 万,在其主界面包含一个广告展示框.该应用在提供娱乐功能时需要玩家的位置信息,因此在运行时申请了位置相关权限.通过利用 WebView 任意代码执行漏洞,攻击者在广告内容中植入读取位置的攻击代码,能够将设备当前位置发送到指定服务器上.盗墓笔记是一款电子书应用,下载量超过 100 万,启动时会展示一个全屏广告框.该应用将用户收藏和阅读过的文章缓存在本地 SDCard 中,在运行时需要申请存储相关权限.通过利用外部存储文件推测漏洞,攻击者在广告内容中植入访问本地存储的攻击代码,能够推测出用户感兴趣的文章类型.

在网络欺诈和恶意扣费方面,针对 nice 应用的攻击能够获得较好的攻击效果.该应用是一款社交类应用,下

载量超过 100 万,能够发布用户的最新动态,并与其他好友保持联系.在应用首页的不同位置嵌入了多个广告框,运行时需要申请读取联系人、打开摄像头和读取短信等高风险权限.通过利用 Web 层和本地层语义误差漏洞,结合数据挖掘技术,攻击者能够获得用户的身份信息,为进一步发动社会工程学攻击和网络钓鱼提供条件.另外,基于短信相关权限,攻击者通过向特定的业务号码发送短信对用户造成直接的财产损失.

Table 5 Four attack cases

表 5 4 个攻击实例

宿主应用				测试设备	攻击结果				
应用名称	分类	安装次数	广告库		窃取隐私	网络欺诈	恶意扣费	拒绝服务	破坏系统
Angry Bird 2	游戏	5 000 000+	Inmobi	Mi 5S	√	–	√	√	–
				M8	√	–	√	–	–
				S6	–	√	–	–	
盗墓笔记	电子书	1 000 000+	亿动广告	Mi 5S	√	√	–	√	√
				M8	√	–	√	–	–
				S6	√	–	–	–	
nice	社交	1 000 000+	AdColony	Mi 5S	√	√	√	–	√
				M8	–	√	√	–	
				S6	√	–	–	–	
WiFi 万能密码	工具	10 000 000+	百分通联	Mi 5S	√	–	–	√	√
				M8	√	–	–	√	√
				S6	√	–	–	–	√

在拒绝服务和破坏系统方面,针对 WiFi 万能密码应用的攻击更加有效.该应用是一款免费的系统工具应用,下载量超过 1 000 万,能够帮助用户连接和共享 WiFi 热点.除了基本的功能以外,应用开发者没有申请过多的权限,因此攻击者只能获得设备相关信息(例如设备 ID、WiFi 状态等),无法发起网络欺诈或者恶意扣费攻击.但是读取各种传感器(例如加速计、陀螺仪等)信息不需要申请任何权限,攻击者利用侧信道攻击漏洞能够妨碍设备的正常使用,并利用 WebView 域控制不严格漏洞卸载设备上已安装的其他应用.

上述分析表明,本文提出的攻击载荷自动生成方法能够根据宿主应用的权限和设备相关漏洞发起相应的攻击,攻击效果显著.另外,在攻击过程中不会出现申请过多权限的情况,从而避免了系统异常的发生,用户更加不容易察觉.

5 防范措施

本文攻击方法能够带来威胁的最主要原因是能够从网络流量中分析得到宿主应用的信息,而这些信息进一步暴露了宿主应用的能力.因此对于移动应用开发者来说,可以借鉴数据发布时隐私保护的方法来隐藏自身信息,从而阻止攻击载荷的自动生成.在广告库获取广告内容时,不要直接加入宿主应用的标识,而是发送处理过的信息.例如,利用 k-匿名[25]技术加入其他 k–1 个应用的标识信息,使得真正的宿主应用无法直接获得.利用泛化技术使用更概括、更抽象的应用数据来代替原始的信息,加大机器学习聚类分析算法识别成功的难度.值得注意的是，在对宿主应用信息进行混淆的过程中,需要考虑对最终服务质量的影响.因为宿主应用嵌入广告的目的是从广告服务商那里获得利益,所以广告服务商需要明确知道哪个应用展示了广告.如果完全去除了宿主应用的信息,那么势必会影响到开发者的利益,也会破坏整个移动广告生态系统.

对于广告服务商来说,为了防止攻击者对广告内容进行非法修改,广告库 AdSDK 可以对获取的内容进行完整性校验.在广告服务端产生广告内容之后,同时生成一段校验码,嵌入在广告内容中,并且保证该校验码无法被修改.当广告库客户端接收到广告内容之后,首先需要提取出校验码和广告内容进行比对.如果在通信过程中广告内容被攻击者修改,那么在客户端校验时会失败,这样,广告库就会知道自身受到了攻击,能够进一步采取防御措施.该方法虽然简单,但是目前大部分广告服务商都没有提供完整性校验功能.

另外,中间人攻击方法为了能够截获宿主应用和广告服务器时间传送的数据,需要依赖 ARP 欺骗或者 DNS 欺骗技术.在移动互联网环境中,ARP 欺骗很难完全防御,只要用户接收和发送 ARP 报文,就有可能会受到虚假

信息的欺骗.传统的配置静态 ARP 缓存的方法对于移动网络环境也不太有效,静态手工维护 MAC 表的方式很难实施.但是,我们仍然可以采取一些方法来降低 ARP 欺骗攻击的几率,例如使用 ARP 服务和 DHCP 服务等.无论是 ARP 欺骗还是 DNS 欺骗,都利用了协议维持信息一致性操作上的缺陷.通过指定局域网内部的一台机器作为 ARP 服务器或者在网关上建立 DHCP 服务器,能够保持网内的机器 IP/MAC 一一对应,从而防止攻击者的冒名顶替.

6 总结与展望

移动应用广告作为一种新型的商业模式,给移动互联网的发展带来了新的机遇,同时,它也给用户的隐私与安全带来了诸多挑战.通过对现有移动广告生态系统的深入分析,本文提出了一种基于宿主权限的移动广告漏洞攻击方法.该方法能够在广告主、广告平台和移动应用都是可信的前提下,通过广告网络发起中间人攻击.首先,本文对广告流量进行分析,从中提取出宿主应用的标识和设备相关信息.宿主应用的标识能够用来得到攻击者的能力上限,设备相关信息能够用来确定攻击者的攻击途径.然后,本文提出了一种基于能力描述语言(CDL)的攻击载荷自动生成方法,利用有限状态自动机生成可行的攻击代码.实验结果表明,本文提出的攻击方案能够达到很好的攻击效果,大量移动广告存在泄露宿主应用标识的问题.最后,针对本文提出的攻击方法给出了一些可行的防御方法.随着攻击者威胁的加剧和移动广告研究的不断深入,相信人们会越来越关注移动广告生态系统中的安全问题.

References:

[1] Rastogi V, Shao R, Chen Y, Pan X, Zou SH, Riley R. Are these ads safe: Detecting hidden attacks through the mobile app-Web interfaces. In: Proc. of the 23rd Network and Distributed System Security Symp. Internet Society, 2016. [doi: 10.14722/ndss.2016.23234]

[2] Demetriou S, Merrill W, Yang W, Zhang A, Gunter CA. Free for all! Assessing user data exposure to advertising libraries on Android. In: Proc. of the 23rd Network and Distributed System Security Symp. Internet Society, 2016. [doi: 10.14722/ndss.2016.23082]

[3] Meng W, Ding R, Chung SP, Han S, Lee W. Thre price of free: Privacy leakage in personalized mobile in-app ads. In: Proc. of the 23rd Network and Distributed System Security Symp. Internet Society, 2016. [doi: 10.14722/ndss.2016.23353]

[4] Book T, Wallach DS. A case of collusion: A study of the interface between ad libraries and their apps. In: Proc. of the 2013 ACM Workshop on Security and Privacy in Smartphones and Mobile Devices. Berlin: ACM Press, 2013. 79–86. [doi: 10.1145/2516760.2516762]

[5] Grace M, Zhou W, Jiang X, Sadeghi A. Unsafe exposure analysis of mobile in-app advertisements. In: Proc. of the 5th Security and Privacy in Wireless and Mobile Networks. Tucson: ACM Press, 2012. 101–112. [doi: 10.1145/2185448.2185464]

[6] Datta A, Tschantz MC, Datta A. Automated experiments on ad privacy settings. Proc. on Privacy Enhancing Technologies, 2015, 1(1):92–112. [doi: 10.1515/popets-2015-0007]

[7] Son S, Kim D, Shmatikov V. What mobile ads know about mobile users. In: Proc. of the 23rd Network and Distributed System Security Symp. Internet Society, 2016. [doi: 10.14722/ndss.2016.23407]

[8] Louw MT, Ganesh K, Venkatakrishnan V. AdJail: Practical enforcement of confidentiality and integrity policies on Web advertisements. In: Proc. of the 19th USENIX Conf. on Security Symp. Washington: USENIX Association, 2010. 371–388.

[9] Zarras A, Kapravelos A, Stringhini G, Holz T, Kruegel C, Vigna G. The dark alleys of Madison avenue: Understanding malicious advertisements. In: Proc. of the 2014 Conf. on Internet Measurement Conf. Vancouver: ACM Press, 2014. 373–380. [doi: 10.1145/2663716.2663719]

[10] Li Z, Zhang K, Xie Y, Yu F, Wang X. Knowing your enemy: Understanding and detecting malicious Web advertising. In: Proc. of the 19th ACM Conf. on Computer and Communications Security. Raleigh: ACM Press, 2012. 674–686. [doi: 10.1145/2382196.2382267]

[11] Liu B, Nath S, Govindan R, Liu J. Decaf: Detecting and characterizing ad fraud in mobile apps. In: Proc. of the 11th USENIX Symp. on Networked Systems Design and Implementation. Seattle: USENIX Association, 2014. 57–70.

[12] Crussell J, Stevens R, Chen H. Madfraud: Investigating ad fraud in android applications. In: Proc. of the 12th Annual Int'l Conf. on Mobile Systems, Applications, and Services. Bretton Woods: ACM Press, 2014. 123−134. [doi: 10.1145/2594368.2594391]

[13] Tuncay GS, Demetriou S, Gunter CA. Draco: A system for uniform and fine-grained access control for Web code on Android. In: Proc. of the 23rd ACM Conf. on Computer and Communications Security. Vienna: ACM Press, 2016. 104−115. [doi: 10.1145/2976749.2978322]

[14] Pearce P, Felt AP, Nunez G, Wagner D. AdDroid: Privilege separation for applications and advertisers in Android. In: Proc. of the 23rd ACM Conf. on ASIA Computer and Communications Security. Seoul: ACM Press, 2012. 71−72. [doi: 10.1145/2414456.2414498]

[15] Shekhar S, Dietz M, Wallach DS. AdSplit: Separating smartphone advertising from applications. In: Proc. of the 21st USENIX Conf. on Security Symp. Bellevue: USENIX Association, 2012. 553−567.

[16] Georgiev M, Jana S, Shmatikov V. Breaking and fixing origin-based access control in hybrid Web/mobile application framework. In: Proc. of the 21st Network and Distributed System Security Symp. San Diego: Internet Society, 2014. [doi: 10.14722/ndss.2014.23323]

[17] NetMate. http://f00l.de/netmate/

[18] Shahshahani B, Landgrebe D. The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon. IEEE Trans. on Geoscience and Remote Sensing, 1994,32(5):1087−1095. [doi: 10.1109/36.312897]

[19] Hartigan JA, Wong MA. Algorithm AS 136: A *K*-means clustering algorithm. Journal of the Royal Statistical Society, 1979,28(1):100−108.

[20] Viennot N, Garcia E, Nieh J. A measurement study of google play. In: Proc. of the Int'l Conf. on Measurement and Modeling of Computer Systems. Austin: ACM Press, 2014. 221−233. [doi: 10.1145/2591971.2592003]

[21] Hopcroft JE, Motwani R, Ullman JD. Introduction to Automata Theory, Language, and Computation. 2nd ed., Addison Wesley, 2003.

[22] Metasploit. http://www.metasploit.com

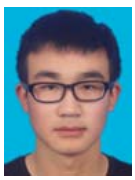
[23] Monkeyrunner for android developer. http://cs.szpt.edu.cn/android/tools/help/monkeyrunner_concepts.html

[24] Operating system market share. https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=10&qpcustomd=1

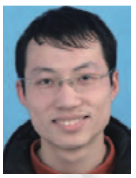
[25] Sweeney L. *k*-anonymity: A model for protecting privacy. Int'l Journal on Uncertainty, Fuzziness and Knowledge-Based Systems, 2002,10(5):557−570. [doi: 10.1142/S0218488502001648]

王持恒(1990－),男,河南漯河人,博士生,CCF 学生会员,主要研究领域为移动安全,恶意软件检测,隐私保护.

陈晶(1981－),男,博士,教授,博士生导师,CCF 专业会员,主要研究领域为网络安全,分布式系统安全.



苏涵(1994－),男,硕士生,主要研究领域为移动安全,漏洞挖掘.



何琨(1986－),男,博士,主要研究领域为网络安全,云安全.

杜瑞颖(1964－),女,博士,教授,博士生导师,主要研究领域为网络安全,密码学.