南京郵電大學
Nanjing University of Posts and Telecommunications

Actions Speak Louder than Words: Entity-Sensitive Privacy Policy and Data Flow Analysis with PoliCheck

汇报人：陈宣衡　指导老师：陈伟

Zhao K, Zhan X, Yu L, et al. Demystifying privacy policy of third-party libraries in mobile apps[C]//2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE). IEEE, 2023: 1583-1595.

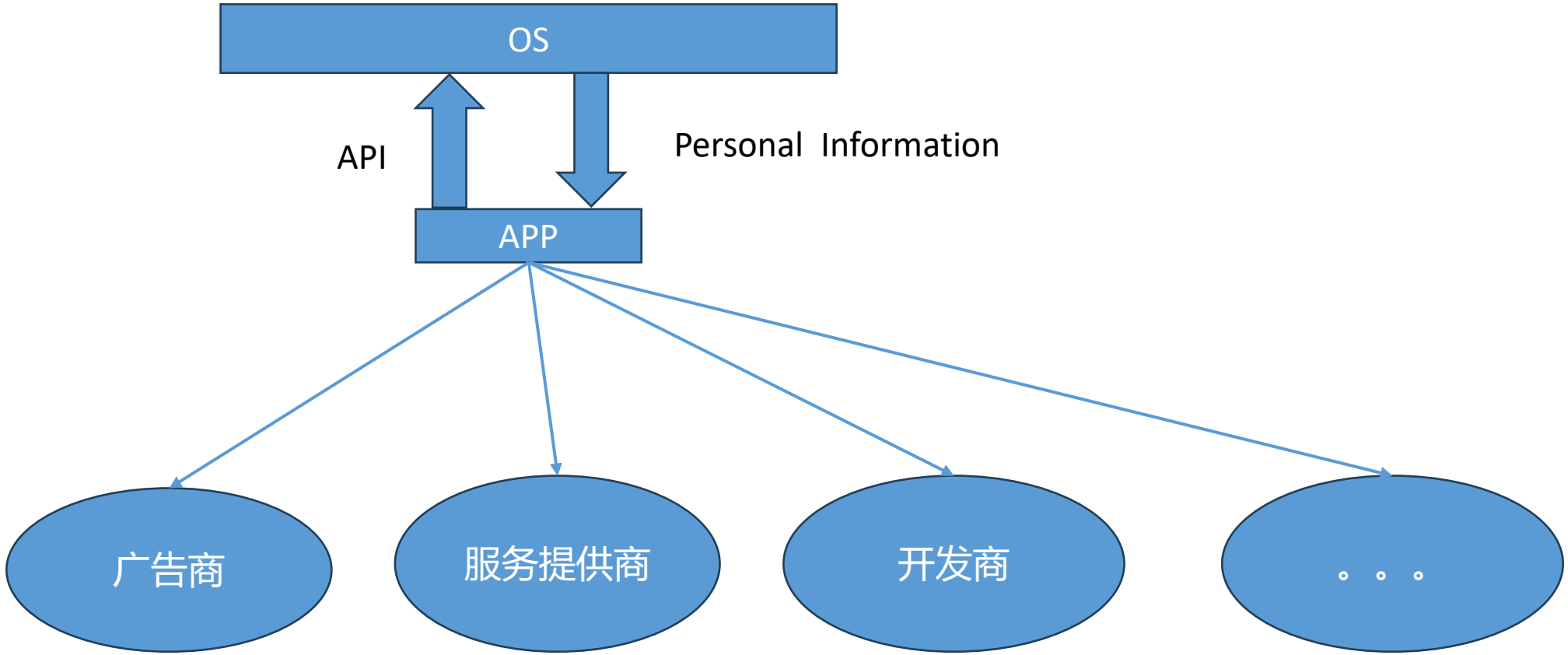
目录

CONTENT

01

背景介绍

背景介绍



背景介绍

➢隐私合规

隐私的概念只是模糊的定义，隐私处于技术、文化和法律的考虑的交叉点。在移动的应用程序的情况下，如果在应用程序的隐私政策中披露数据收集和共享，则数据收集和共享通常被认为是（法律上）可接受的。

When you launch any of our applications, we collect information regarding your device type, operating system and version, carrier provider, IP address, Media Access Control (MAC) address, International Equipment Mobile ID (IMEI), whether you are using a point package, the game version, the device's geo-location, language settings, and unique device ID.

背景介绍

➢TPL

功能

越来越多的开发人员倾向于使用许多现成的第三方库来促进开发过程， Android TPL提供丰富的功能（例如，用户数据分析和广告推荐）

存在问题

TPL可能会在未经用户同意的情况下滥用权限访问用户的个人身份信息
开发人员不检查TPL的数据使用情况
开发人员可能无法在其隐私政策中清楚地描述TPL的数据使用情况

法规要求

CCPA第1798.120条声称“如果将消费者的个人信息出售给第三方，企业应通知消费者”。

网络安全实践指南.移动的互联网应用软件开发工具包（SDK）使用安全指南（SGSDK）”要求SDK以清晰、易懂、合理的方式向App披露SDK处理个人信息的范围、目的和规则。SDK收集和使用个人信息的实际行为应与公开文件中的声明一致。”

背景介绍

➢先前工作

PoliCheck：执行动态分析以获取流量，从而分析应用程序的数据使用情况。然后，应用数据和实体依赖树分析从隐私政策中提取数据使用语句，并设计规则来识别冲突。

PPChecker：检查应用程序隐私政策的可信度以及应用程序行为与其隐私政策之间的一致性，而不考虑最新的法规。

缺点：不能识别TPL的功能或分析TPL隐私政策

背景介绍

➢本文贡献

本文提出了一个新的系统ATPChecker来分析Android TPL的合规性。ATPChecker使用静态分析来识别TPL的用户数据访问行为和主机应用与TPL的数据交互，并使用自然语言处理技术来分析TPL和主机应用的隐私政策。结合字节码分析和隐私政策分析的结果， ATPChecker确定TPL以及TPL在主机应用中的使用是否符合规定。

为了评估ATPChecker的性能并促进该领域的进一步研究，本文构建了一个隐私数据集，其中包括187个TPL的二进制文件及其隐私政策，以及642个主机应用程序及其隐私政策。本文的数据集包含TPL的隐私政策以及相关的数据访问声明。本文的数据集包括主机应用程序的隐私政策，其中包含与主机应用程序与TPL交互相关的标签。

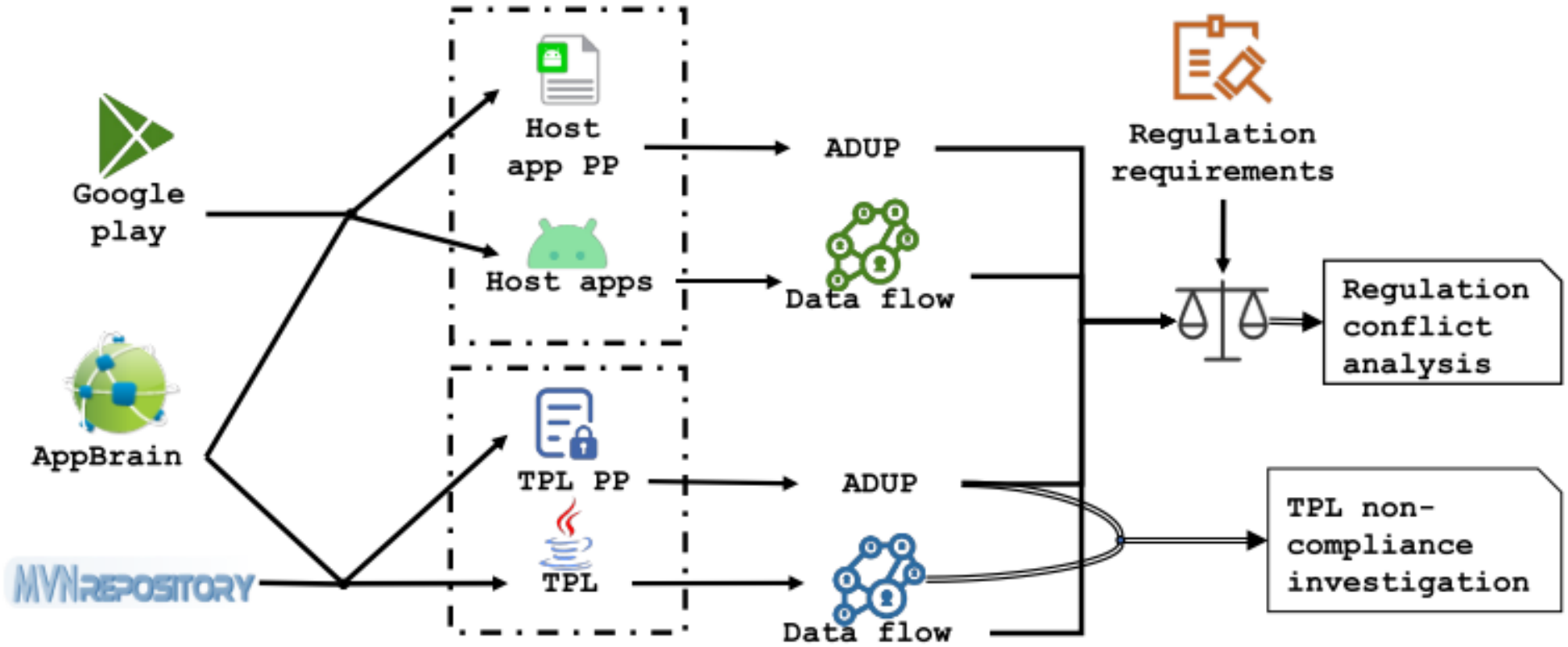
ATPChecker发现，超过31%的TPL没有提供隐私政策，47%的TPL的隐私政策隐藏了数据使用情况。 ATPChecker发现，超过65%的主机应用程序违反了明确披露与TPL的数据交互的法规要求。

02

研究方法

研究方法

➢框架



Abstract data usage patterns (ADUP)： {data entity, action, {data type}, {data recipient}, {neg}}.
“We share your personal information with our service providers”
{app, share, {personal information}, {service provider}, false}.

研究方法

➢确定TPL的数据使用情况

ATPChecker使用方法调用图（FCG）来跟踪方法之间的PI（personal information）流，分析发现，soot不能有效地发现TPLs的主要功能和入口点，以构建有效的FCG，于是提出以下方法来优化FCG构造：

1、ATPChecker遍历每个类和方法。由于apk主要通过调用TPL的公共方法来使用TPL，因此将TPL的公共方法扩展到入口点集，以优化FCG构造

2、手动抓取PI相关的API 。基于这些API，ATPChecker迭代所有语句并使用目标变量定位DOI（data of interest）。执行过程间和过程内数据流分析，以识别所有PI相关语句。对于没有官方API的DOI，如电子邮件和密码，使用关键字匹配方法来识别潜在的数据泄漏。

TABLE I: Tracked data types in static analysis.

Data Type
Ad ID, username, password, name, location, contact, phone number, email address, IMEI, Wi-Fi, MAC address, GSF ID, Android ID, serial number, SIM serial number

研究方法

➢确定TPL的数据使用情况

过程间分析

```
Def BackwardAnalysis(m_t, s_t, v_t):
    defs = getDefsOfAt(v_t, s_t, cfg)
    for d in defs do
        src_var = get the signature of v_t in d
        caller_list = find methods that invoke m_t
        for m in caller_list do
            src_stmt = statements that invoke m_t in m
            src_v = variables in src_stmt that correspond
             to v_t
            if src_v is Variable then
                BackwardAnalysis(src_v, src_stmt, src_m)
            else if src_v is Constant then
                store (v_t, src_stmt) in DF
```

过程内分析

```
Def IntraMethodVarAnalysis(m, stmt, var):
    uses = find statements that use var in method m
    for u in uses do
        store(var, u) in DF
        ForwardAnalysis(m, u, var)
```

```
Def ForwardAnalysis(m, stmt, var):
    tar_stmt = find all statements that contain stmt
    for s in tar_stmt do
        store (var, s) in DF
        if s contain invoke statements then
            tar_m = method that invokes stmt in s
            tar_v = variable in tar_m that correspond to
             var
            ForwardAnalysis(tar_method, s, tar_v)
```

研究方法

➢识别TPL隐私政策中的数据使用声明

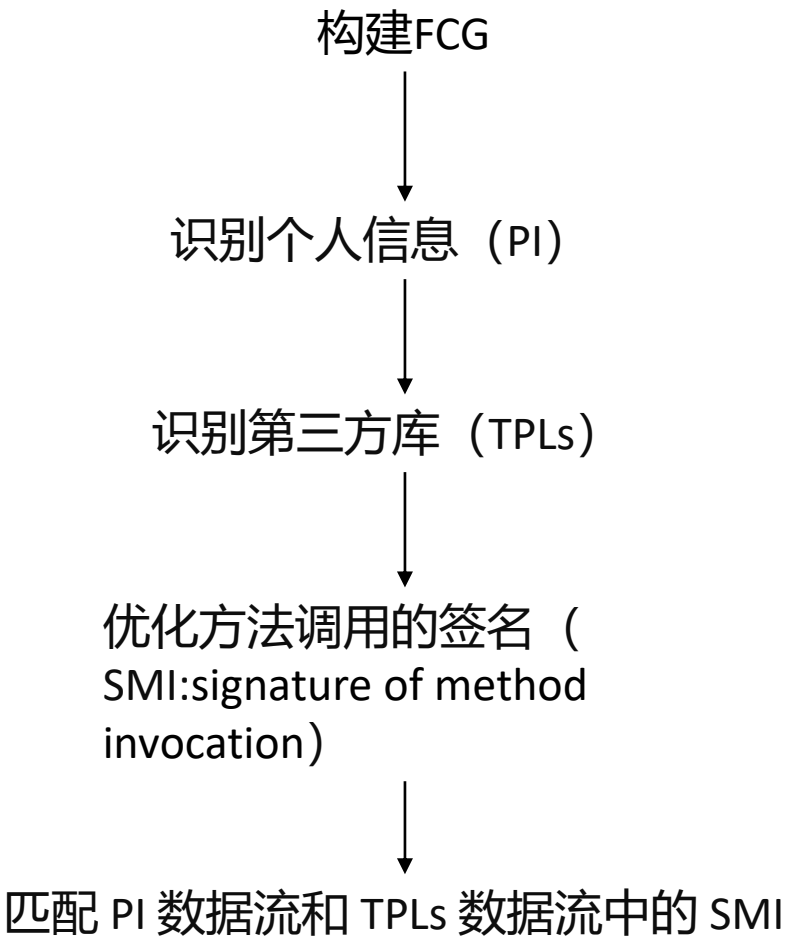
1、隐私政策预处理
下载html→转换为纯文本→消除特殊字符、合并相关句子

2、数据访问提取
识别数据动词→识别数据实体→识别数据参与者

Action Type	Keywords
Collect	access, check, collect, gather, know, obtain, receive, save, store, use
Sharing	accumulate, afford, aggregate, associate, cache, combine, convert, connect, deliver, disclose, distribute, disseminate, exchange, gather, get, give, keep, lease, obtain, offer, post, possess, proxy, provide, protect against, receive, rent, report, request, save, seek, sell, share, send, track, trade, transport, transfer, transmit

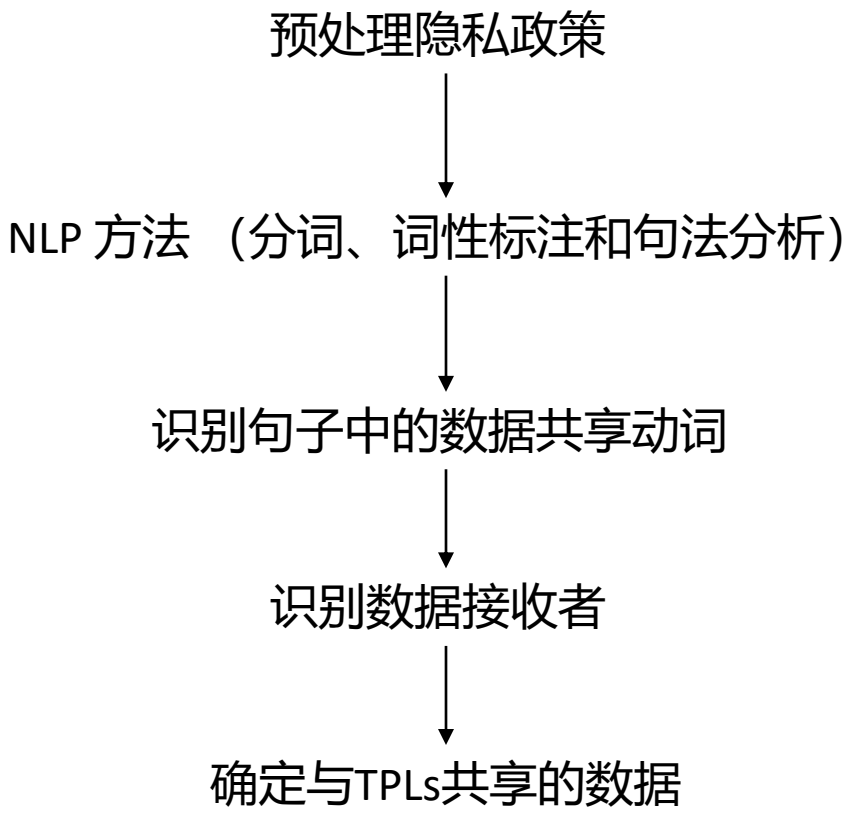
研究方法

➢提取主应用程序与TPL的交互



研究方法

➢确定主应用程序隐私政策中的TPL使用声明



研究方法

➢TPL隐私合规调查

规范性分析确定TPL是否提供隐私政策。

合法性分析确定了TPL的行为与隐私政策声明之间的冲突。

隐私违规可传递性分析检查其隐私政策违反法规要求的TPL对其他TPL或应用程序的影响程度。

03

方法评估

方法评估

➢TPLs的规范性分析

数据集

458个TPL，其中包括141个广告网络，25个社交库和292个开发工具

方法

通过访问AppBrain上每个TPL信息页面提供的主页来收集这些TPL的隐私政策

结果

141个广告网络TPL中有21个，25个社交TPL中有10个，292个开发工具TPL中有180个，占31%，不提供隐私政策网站

方法评估

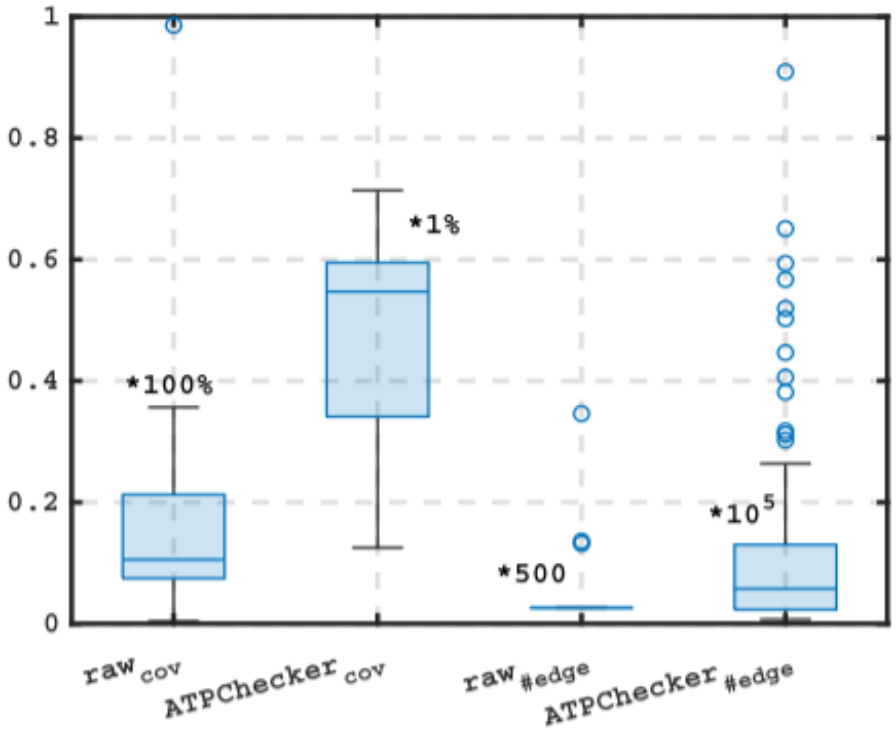
➢TPLs的合法性分析

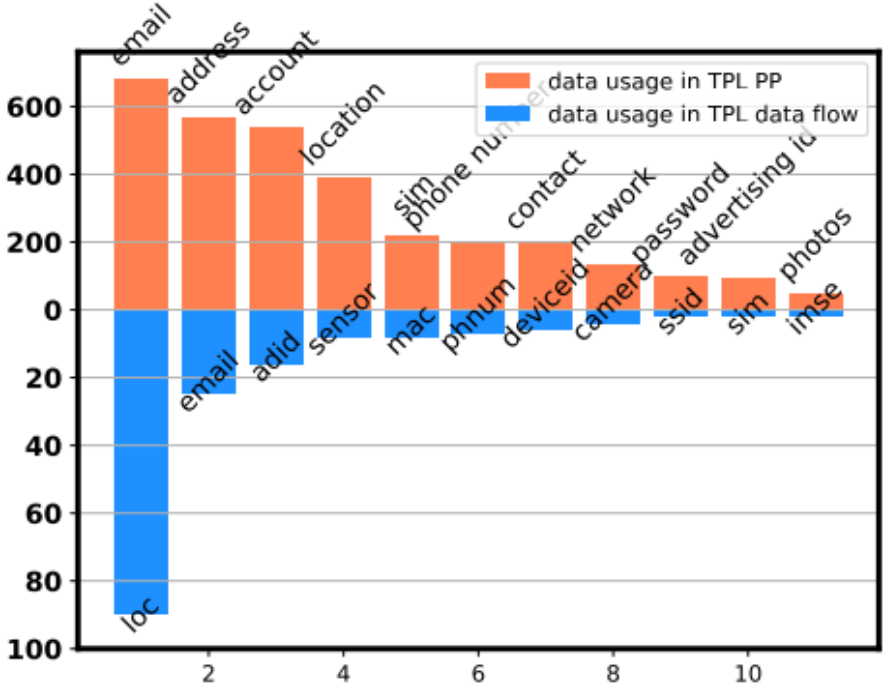
数据集
187个不同的TPL，包括87个“aar”和100个“jar”文件。其中45个广告网络，132个开发工具库和10个社交库。

TPL隐私政策分析

TPL数据使用分析

通过过检查数据流中的PI是否也在ADUP中提及来识别有多少TPL符合规定





38家TPL中有18家（47.4%）违反了法规要求

方法评估

➢TPLs的合法性分析

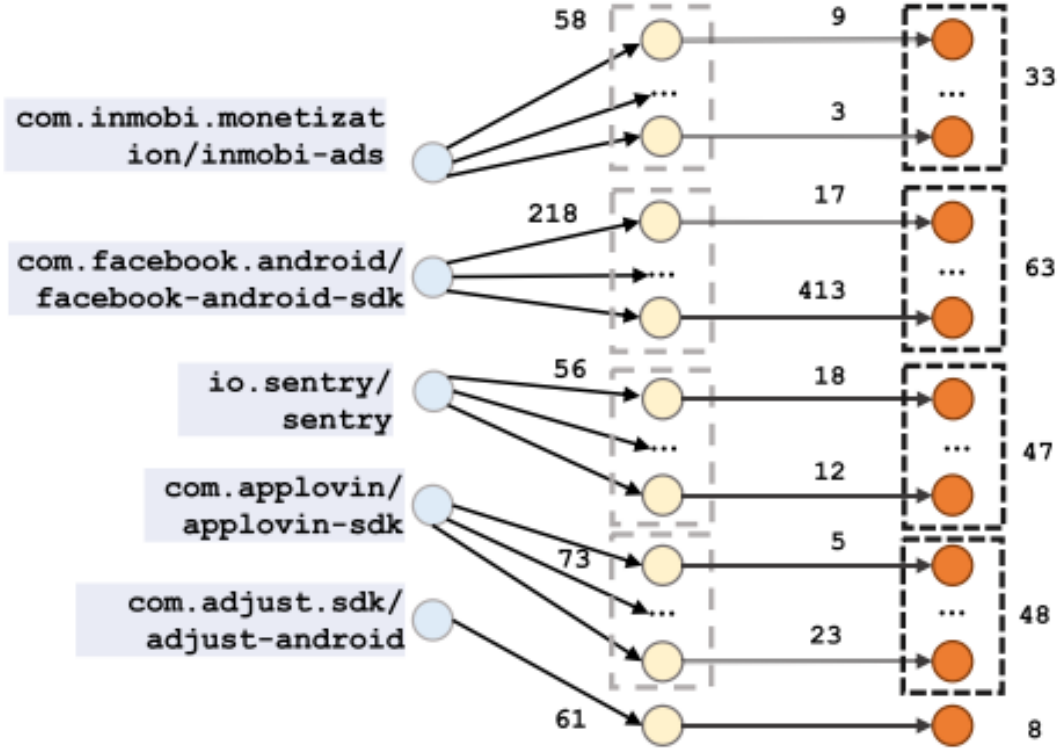
方法：结合数据流分析和隐私政策分析的结果，通过过检查数据流中的PI是否也在ADUP中提及来识别有多少TPL符合规定

结果：38家TPL中有18家（47.4%）违反了法规要求

方法评估

➢TPLs的合法性分析

TPL不合规行为危害性分析



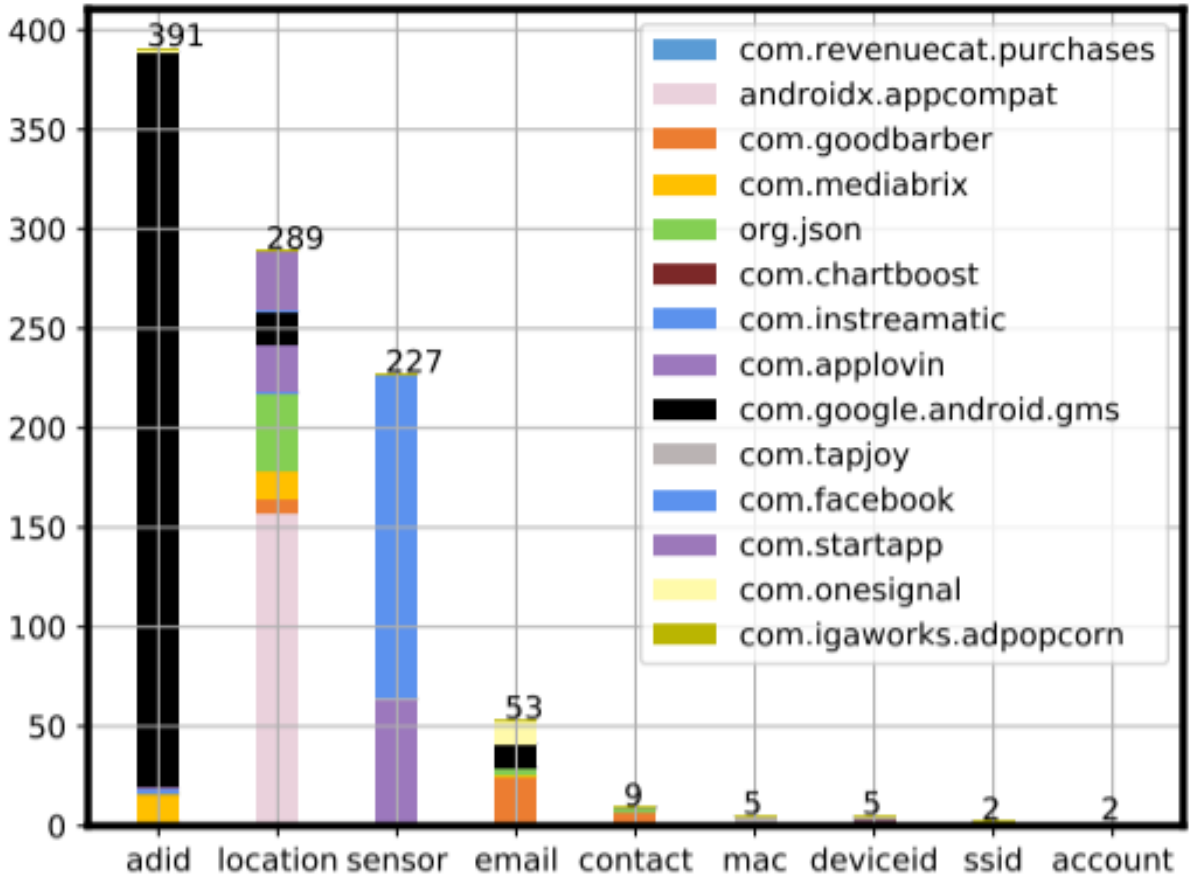
方法评估

➢主应用程序行为分析

数据集

ATPChecker分析了641个不同的应用程序。由于计算资源的限制和flowdroid的局限性，成功分析了459个应用程序

结果
超过47.9%（220/459）的主机应用程序与TPL共享PI。ATPChecker通过统计应用数据流中PI相关调用次数来衡量可疑行为，并追踪了973次与TPL的数据共享行为，发现超过40.2%的追踪者将用户的广告ID与“com.google.android.gms”等TPL共享。



方法评估

➢主应用程序隐私政策的合法性

应用程序是否在其代码中集成了TPL，而没有在隐私政策中披露?

应用程序是否在没有描述的情况下与TPL进行数据交互?

	TP	TN	FN	FP
TPL_list	95	0	0	9
TPL_data	223	/	/	12
	Accuracy	Precision	Recall	F1
TPL_list	0.91	0.91	1	0.95
TPL_data	/	0.95	/	/

04

总结与思考

总结与思考

使用静态分析来识别TPL的用户数据访问行为和主应用与TPL的数据交互
使用自然语言处理技术来分析TPL和主机应用的隐私政策
确定TPL以及TPL在主应用中的使用是否符合规定

缺点
1、基于静态分析工具，如soot和flowdroid，不能处理TPL的某些动态行为（例如，反射，动态类加载）。
2、仅限于预定义的模式和字符串匹配，用于识别收集的隐私政策中的语句和TPL使用，可能会导致一些不正确的匹配。

改进方法
1、可以加入动态分析或者流量分析来降低误报率。
2、对于隐私政策处理可以利用一些成熟的工具如policylint。

南京郵電大學
Nanjing University of Posts and Telecommunications

敬请各位老师批评指正