移动设备加密流量的用户信息探测研究展望

张腾飞，余顺争
（中山大学计算机学院，广东 广州 510006）

摘 要：移动设备加密流量分析可以用主动或被动的方式获取多种类型的用户信息，为网络安全管理和用户隐私保护提供保障。重点分析、归纳了用户信息探测所涉及的数据采集、特征选择、模型与方法以及评价体系的基本原理和关键方法。总结了现有方案中存在的问题，以及未来研究方向和面临的挑战。

关键词：移动设备；加密流量分析；网络安全；隐私保护；信息探测
中图分类号：TN92
文献标识码：A
DOI: 10.11959/j.issn.1000−436x.2021040

Research prospects of user information detection from encrypted traffic of mobile devices

ZHANG Tengfei, YU Shunzheng
School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China

Abstract: Encrypted traffic analysis of mobile devices can obtain multiple types of user information in an active or passive way, which provides protection for network security management and user privacy protection.The basic principles and key methods of data collection, feature selection, models and methods, and evaluation systems involved in these user information detection were analyzed and summarized. The problems in the existing projects were summarized, as well as the future research directions and challenges.

Keywords: mobile device, encrypted traffic analysis, network security, privacy protection, information detection

1 引言

第 45 次《中国互联网络发展状况统计报告》[1]显示，截至 2020 年 3 月我国手机网民规模达 8.79 亿，我国网民使用手机上网的比例高达 99.3%。移动设备产生的巨大网络流量对网络管理、网络服务质量等都提出了新的要求。

移动设备中往往存储了大量的用户个人数据和通过各类 App 产生的海量级数据，这些数据大多涉及个人隐私，从而导致越来越多的攻击者开始分析移动设备的网络流量，并试图从中挖掘有用的信息。因此，移动设备敏感数据的安全问题显得尤其突出[2]。除了需要设计有效的数据、位置和身份隐私保护方案之外[3-5]，如何将传统的隐私保护方案与网络流量分析相结合，使其在多样化的服务环境中实现用户隐私保护，是当前的一个重要研究方向。

当前，越来越多的移动应用开始使用加密技术为数据传输提供安全保障，这给基于网络流量分析的用户信息探测带来了挑战，主要表现在以下几个方面：1) 流量加密后负载特征和流量统计特征都发生了改变；2) 之前非加密的流量分析方法很难应用在加密的流量分析上，如深度包检测（DPI, deep

收稿日期：2020−07−14；**修回日期**：2020−11−03
基金项目：国家自然科学基金资助项目（No.U1636118，No.61972431）；广东省科技计划项目重大专项基金资助项目（No.2017B030306015）；广东省应用型科技研发专项基金资助项目（No.2017B010125003）
Foundation Items: The National Natural Science Foundation of China (No.U1636118, No.61972431), The High Technology Research and Development Program of Guangdong (No.2017B030306015), Special Project on the Applied Technology Research and Development of Guangdong (No.2017B010125003)

packet inspection）技术[6]；3) 加密流量常采用协议混淆进行伪装[7]，如随机端口和各种复杂的加密算法。在不破解加密数据的情况下，攻击者仍然可以窥探用户的隐私数据[5, 8-16]。目前，针对移动设备的加密流量分析有两大方向：一是在传统的网络流量分析方法上进行改进；二是开发专门应用在移动设备上的加密流量分析方法。

有研究者开始将网络流量分析技术应用在移动设备的安全和隐私分析上。文献[17-18]综述了当前移动设备的网络流量分析的研究成果，但这些成果大多集中在手机应用（下文简称 App）分类、设备指纹、操作系统识别、用户动作识别、服务（或业务）分类（如语音、在线视频直播、视频观看及下载）、网页浏览、恶意应用检测等方面。本文侧重于综述如何在各类网络节点的网络流量中识别不同用户及其应用的使用习惯、操作行为、社会关系等用户个人信息。这些信息对精细用户画像和网络安全管理有很大帮助。其现实意义主要表现在以下几个方面。

1) 从加密通道下不同应用和协议的流量中，发现隐藏的攻击行为、恶意行为和非法行为。这对掌握网络安全态势、发现网络异常和维护网络安全管理都有重要意义。

2) 特殊协议或私有协议在加密通道掩盖下的网络行为的识别和分析，对维护网络安全管理和提高安全防御具有重要的支撑作用。

3) 对恶意网络流量和非法通信的识别和记录有助于犯罪痕迹调查。

根据网络流量识别的目的，研究者通常从协议、应用、服务等角度对网络流量进行深入分析和识别。根据移动设备流量加密识别的需求，可将识别类型分为 App 识别，用户行为识别，个人标识信息泄露检测，使用研究与操作系统识别，社会关系以及输入法语言、国籍、健康状况等信息识别。

考虑到已有研究的现状和急需解决的用户数据安全与隐私问题，本文主要对当今用户信息泄露探测和隐私保护领域密切相关的研究展开论述，包括个人标识信息、服务使用、用户动作识别，其中服务使用和用户动作都是用户在 App 上的操作行为，因此统称为 in-App 操作信息。

2　个人标识信息探测

移动设备上大量关于用户的个人敏感信息（PII, personal identifiable information）通常分为以下 4 种类型。

1) 移动设备硬件相关信息，包括国际移动设备标志（IMEI, international mobile equipment identity）、Android 设备 ID 与介质访问控制（MAC, medium access control）地址、国际移动用户标志（IMSI, international mobile subscriber identity）和 SIM 序列号。

2) 凭据信息，包括用户名、密码。

3) 人口统计信息，包括姓名、性别、出生日期、居住地址、电话号码和电子邮件地址等。

4) 地理位置信息，包括 GPS 定位和邮政编码等。

当前对 PII 探测的方法分为静态分析、动态分析、静态分析与动态分析结合，以及网络流量分析。但静态分析需要人工分析，耗时耗力，不便于大规模检测；动态分析受程序保护机制影响，不能正确识别污点走向。因此，如何从网络流量分析角度获取这类信息成为一个值得关注的研究点。已有研究显示[19]，移动设备上安装的 App 更容易产生被识别的流量特征，例如，平均数据分组长度、数据分组长度的分布以及平均数据分组到达时间间隔，其比使用浏览器访问相同的服务更容易泄露隐私数据[9, 20]。然而，在网络中被动监听流量受加密影响，只能获取数据分组头部信息，缺乏语义信息。因此，通过对移动设备上各类 App 的网络流量进行主动测量分析，可以判定是否存在用户敏感信息泄露以及信息泄露的程度。为了阻止对隐私泄露提供帮助，常用方法是把用户个人信息进行替换处理或加密混淆。

2.1　基于虚拟专用网（VPN, virtual private network）的模式匹配分析方法

基于VPN的网络监控工具使用tun接口来捕获移动设备上的网络流量，然后根据模式匹配方法从网络流量中探测用户敏感信息。该方法需要在手机上安装定制的 VPN 客户端进行流量采集，而不需要定制操作系统和获取系统根权限，因此有利于大规模部署。基于 VPN 的模式匹配分析方法流程如图 1 所示。其核心方法是利用 Android 的 VPN 接口开发一款基于中间人（MIMT, man-in-the-middle）原理和传输层安全（TLS, transport layer security）协议代理技术的 App，以实现监听移动设备上 App 的网络流量，然后利用字符串匹配或正则匹配进行 PII 探测。

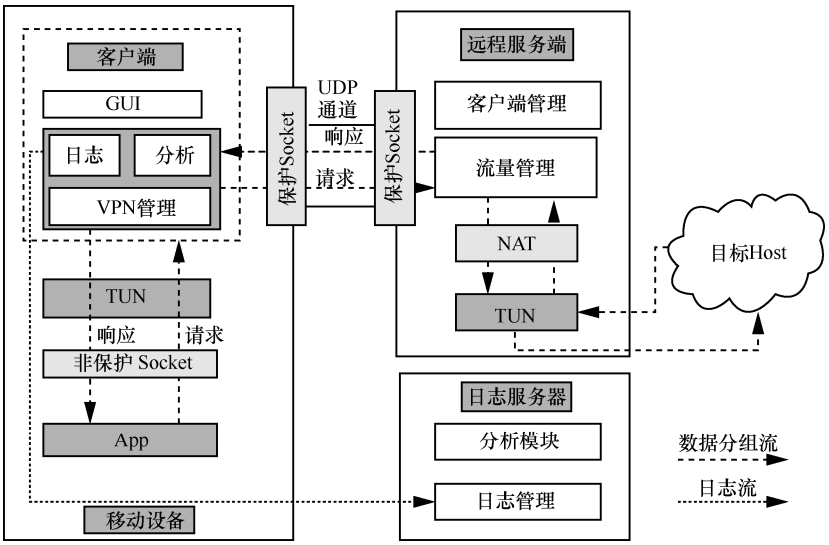


图 1　基于 VPN 的模式匹配分析方法流程

1) 基于字符串匹配的分析方法

文献[21]利用基于中间人原理的VPN客户端和远程 VPN 代理，将流量传输到远程 VPN 服务器进行分析。在进行 PII 泄露检测前，需要先根据设备信息、网络信息、操作系统指纹和历史地址位置等信息构建一张用户敏感信息表，然后利用字符串匹配算法 Aho-Corasick 扫描每条网络流，以判断是否存在用户敏感信息泄露。与其原理相似，文献[22]将远程服务端的功能剔除，仅用客户端分析 PII 泄露。文献[23]在 PrivacyGuard[24]的基础上设计了一种高效的近似字符串匹配算法 Boyer-Moore，能够对移动设备上所有 App 的网络流量进行扫描，实现对敏感和包含 PII 的数据分组中目标字符串的精确匹配，显著提高了 PII 搜索效率。当发现 PII 时，可以向用户提供允许或阻止数据分组的选择，但该方法只能对以字符串编码的 PII 泄露进行检测，而对其他类型的信息表示方法无效。

文献[25]在流量监管系统 AntMonitor[21]基础上改善了流量路由机制，移除了 VPN 服务器，实现了对流量细粒度的、大规模的无源收集，所需的资源更少，处理性能接近实时，对用户透明，几乎不影响用户正常的使用体验，其性能优于 Haystack[22]和 PrivacyGuard[24]。文献[26]自定义加密流量上下文信息（如 PII 类型、目标域、应用类别/功能、背景/前景、使用加密和纯文本），利用改进的 Antmonitor[25]进行 PII 泄露检测，不仅能够揭示 PII 泄露程度和模式，还可以发现一些未知的 PII 类型。

文献[27]首先将结构化的 HTTP 请求头的键值对构建成哈希签名，用来表示请求内容，并将其传送到远程服务器端，与文献[28]相比，该方法没有显示任何实际的用户数据，减轻了隐私和安全的风险；然后，使用一种 crowd-based（或称为 crowd-sourced）方法，匿名合并来自同一设备的不同用户的签名，根据“仅来自特定设备的数据流中反复出现的键值可能是 PII，而许多设备都出现的键值则不是 PII”这一原理，以自适应和可伸缩的方式检测 PII。这种方法既不需要标记网络流，也不需要任何用户交互。

2) 基于正则匹配的分析方法

文献[24]通过自定义的 PII 检测类插件（如 Location Dection、Phone State Dection、Contact Dection）来检测移动用户的 PII 泄露情况。其中，Location Dection 插件为每个位置服务提供者登记一个 Location Listener（Android 的一个概念，提供位置信息），来判定位置信息的泄露；Phone State Dection 插件通过使用相应的 Android 应用程序接口（API, application programming interface）调用获取设备硬件相关信息，然后在网络流中查找这些值；Contact Dection 插件使用正则匹配来检测人口统计信息的泄露情况。当 App 自身在传输前对敏感信息进行加密时，会使 PII 检测失败。当前，该方法只对以 RSA（Rivest, Shamir, Adleman）方式交换 TLS 的证书有效。

通常基于正则匹配的分析方法有以下几点限制。

① 当移动设备阻止 App 写入受信任的根存储时，会无法安装自定义的 TLS 证书，如果 App 使用证书锁定，也会影响 TLS 代理，导致无法拦截加密流量。

② 在给定的时间内，Android 系统只允许一个 App 使用 VPN 接口，因此，如果另一个 App 正在使用该接口，系统将无法运行。

③ 已有的方法不能处理网络层加密的流量，例如，对网络层分组通过 IPSec 加密处理。

④ 稳健性较差。当目标字符串有微小改动时，字符串匹配方法的匹配准确率会发生波动。

⑤ 受限于 App 的 UNIX 文件描述符，在运行时为每个流打开一个新的套接字的 VPN 应用程序进程，因此，需要谨慎地处理套接字资源，否则会造成资源枯竭，系统卡死。

⑥ 只能对基础编码和标准哈希算法模糊化的字符串进行 PII 泄露的探测，而无法应对自定义 App 加密方法、非标准化编码等模糊策略处理过的 PII。

2.2　基于 VPN 的机器学习分析方法

出于保护用户隐私的考虑，一些 App 使用字符串模糊和编码策略对 PII 进行保护，这使基于 VPN 的模式匹配分析方法的探测效果大打折扣。因此，基于 VPN 的机器学习分析方法逐渐被应用到 PII 探测中，其流程如图 2 所示，通常分为离线训练模型和在线处理模型两部分。离线时，使用标记的流量训练分类器，输出 PII 预测模型；在线时，使用模型对流量进行 PII 泄露检测，将可疑的 PII 以 UI（user interface）显示给用户，让用户判定 PII 预测是否准确，是否阻止这些信息泄露，然后将用户反馈结果添加到训练集中，再次训练分类器、更新模型。这类方法除了基于 VPN 和 TLS 代理技术拦截流量的共性问题外，还有以下几点特性。第一，除文献[28]外，其他方法只在移动设备端处理数据，不需要任何远程 VPN 服务器。第二，对已知类型 PII 进行模式匹配，对未知类型 PII 使用分类算法，能够检测动态变化的 PII 泄露信息和不在先验知识列表中的 PII 类型。

1) 基于监督学习算法

文献[28]利用该团队之前研发的跨平台网络流量收集与分析系统 meddle[29]，通过 VPN 将所有移动设备的流量重定向到代理服务器；然后在远程服务器中，即使在没有任何 PII 的先验知识情况下，通过用户对疑似 PII 反馈策略和机器学习方法构建分类器来推测未知类型的 PII 泄露。该方法不能检测经过混淆方法处理后的 PII 泄露，需要将数据传输到远程服务器进行训练，有一定的安全隐患，且违反用户隐私数据本地保护策略。文献[30-31]的方法与其最相近，对未知类型 PII 识别时，利用多标签分类算法（二值相关和决策数），实现在线实时检测 PII 泄露情况和类型。由于是对网络中传输的数据分组内容进行检测，因此可以为用户提供透明度并实时控制其个人信息。该系统是首个实现分布式 PII 检测的方法。

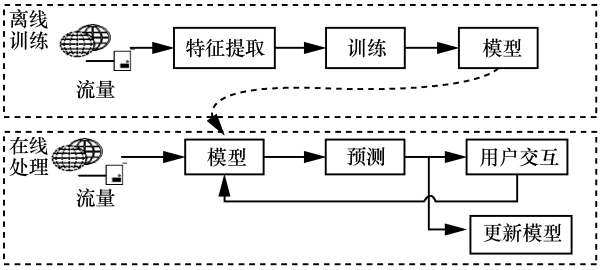


图 2　基于 VPN 的机器学习分析方法流程

文献[32]从网络流量直接推断用户 PII，其原理是利用以下 2 个观察：①开发人员命名约定会经常提示数据收集的目的，例如 URL 路径；②外部知识、App 元数据和关于域名的信息可以用来推断不同请求行为的流量。将每个请求体解析为键值对，并结合使用监督学习和 bootstrapping 自然语言处理（NLP, natural language processing）方法推断每个键值对的数据类型和数据收集目的，根据数据收集与服务功能匹配情况判定 PII 泄露程度。文献[33]利用 C5.0 决策树算法和基于众包的技术构建分类器，用于检测网络流量中任何形式的 PII 泄露，并使用泛化技术来降低隐私敏感性。此外，通过使用 API hook 技术，能够动态定制每个 App 的数据，对提供给 App 的信息进行细粒度控制，可以在运行需要时修改访问权限。

2) 关联挖掘方法

关联挖掘方法示例如图 3 所示。为了挖掘简单字符串搜索和规则匹配无法显示的隐藏信息，如图 3(b)中黑色字体加粗部分的键值对 bssid=28%3AC2%3ADD%3A4D%3AFC%3AA9 是一个 MAC 地址，而不是 Basic Server Set ID，文献[20]提出了基于请求数据分组头部先定义键集合，通过大量 App 流量提取键值对，如图 3(a)和图 3(b)中黑色加粗字体所示；然后，分别构建键集合关联的值集合，同理构建值对应的键集合；最后，进行迭代匹配的关联挖掘。该

方法假设在会话中单个终端上的个人标识符、设备标识符和位置信息等 PII 不会被改变。利用键与值之间的关联，发现更多不同键之间的间接关联，如图 3(c) 所示，键“latitude”最终匹配一对键值“cityid=24”，从而扩大隐私泄露检测的覆盖范围。

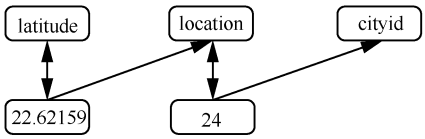
GET
commdatav2?cmd=51& **app_version_name=653&app version_build=&so_name=p2p&so_verV0.0.0.0&app_id= 248&sdk_version=V4 1.248 1730&imei=868129022933673&imsi=460239181213 29&mac=cc:df:3a:13:50:66&numofcpucorc=8&cpufreq=1363**&cpu
Host: mcgi.v.qg.com
User-agent: Apache-HttpClient/UNAVAILABLE(java1.4)
Full request URI: http://mcgi.v.qg.com/
commdatav2?cmd=51&app_version_name=6.5.3&app_version build-0&so_name-p2p& so vel=V0.0.0.0& app id=248& sdk versio=V4.1.248.1730&imei=868129022933673&imsi=460023918121329&mac=es:df:3a:f3:50:6
HttpreQuest 1/1

(a) GET请求

POST
https://fxXXXxx.com/pgc/ma/?cityid=361&tengxun_new&aid=
l&max_behottime=HTTP:/
1.1
Host: lfxXXXXX.com
Connection: keep-alive
Accept: application/json, text/javascript
User-Agent: Mozilla/5. 0(Linux: Android 4.4.2; MI 6 Build/NMF26X)
Referer https://fxXXXxx.com/user/profile/native_index
?user_id=935892161 &is_following=0

data=%7B%22bssid-**28%3AC2%3ADD%3A4D%3AFC%3AA9**%2C%22location%22%3Atrue%2C%22phone%22%3Atrye%2C%22storage%22%3Atrue%7D%2C%22night_nowif2%3A%2C%22apn_notify%22%3A1%2C%22switch_domain%22%3AO%2C%22video_nowifi_notice_mode%22%3A0%2C%22refresh_mode%22%3A0%2C%22comment_mode%22%3A

(b) POST请求



(c) 关联挖掘

图 3 关联挖掘方法

虽然这类方式可以检测出一些未知 PII，但与 2.1 节基于正则匹配的分析方法一样存在 6 种主要限制。

针对基于 VPN 的 PII 探测方法，表 1 对客户端与服务器配置需求、未知类型、实时性、探测能力以及稳健性等指标进行了对比描述。其中，未知类型指对未知类型 PII 的探测能力，目前只有基于 VPN 的机器学习分析方法能够探测出一些未知类型的 PII；实时性指该方法是否能够实时或接近实时在线进行 PII 探测；探测能力指能够探测 PII 类型的种类以及数量；稳健性指对使用未知混淆策略模糊化的字符串进行 PII 探测的能力。

2.3 基于差分分析的方法

基于 black-box 差分分析的方法步骤如下：首先，建立一个 App 网络行为的基线；然后，修改移动设备的 ID 和位置等 PII，收集在相同实验环境下的多次执行的网络流量和上下文信息；最后，观察网络流量中的偏差（差分分析和风险分析）来检测 PII 泄露[34]。该方法的流程如图 4 所示。基于差分分析的方法的主要挑战之一是消除 App 不同执行之间的所有不确定性来源，这样才能可靠地将输出的变化属性归结为输入的变化，并确认是否存在 PII 泄露。文献[35]观察发现网络行为中的非确定性可以被解释和消除，使通过差分黑盒分析进行隐私泄露检测成为现实，并基于此原理开发了一款开源的移动设备 PII 泄露检测工具 Agrigento，它能够适应编码、格式化、加密等混淆技术，或在隐私信息泄露前对其执行的其

表 1 基于 VPN 的 PII 探测方法对比

方法	提出年份	文献	客户端	服务器	未知类型	实时性	探测能力	稳健性
字符串匹配	2015	文献[21]	✓	✓	☆☆☆☆☆	×	★★☆☆☆	★★☆☆☆
字符串匹配	2015	文献[22]	✓	×	☆☆☆☆☆	×	★★☆☆☆	★★☆☆☆
字符串匹配	2020	文献[23]	✓	×	☆☆☆☆☆	×	★★★☆☆	★★★★☆
字符串匹配	2016	文献[25]	✓	×	☆☆☆☆☆	✓	★★☆☆☆	★★★☆☆
字符串匹配	2020	文献[26]	✓	×	☆☆☆☆☆	×	★★★★☆	★★★☆☆
字符串匹配	2017	文献[27]	✓	✓	☆☆☆☆☆	×	★★☆☆☆	★★★☆☆
正则匹配	2015	文献[24]	×	×	☆☆☆☆☆	×	★★☆☆☆	★★☆☆☆
监督学习	2016	文献[28]	✓	✓	★★★★☆	×	★★★★☆	★★★★☆
监督学习	2018	文献[30]	✓	✓	★★☆☆☆	✓	★★★★☆	★★★☆☆
监督学习	2018	文献[31]	✓	×	★★☆☆☆	✓	★★★★☆	★★★☆☆
监督学习	2018	文献[32]	✓	×	★★☆☆☆	×	★★★★☆	★★★☆☆
监督学习	2018	文献[33]	✓	×	★★☆☆☆	×	★★★☆☆	★★★☆☆
关联挖掘	2019	文献[20]	✓	×	★★★☆☆	×	★★☆☆☆	★★★☆☆

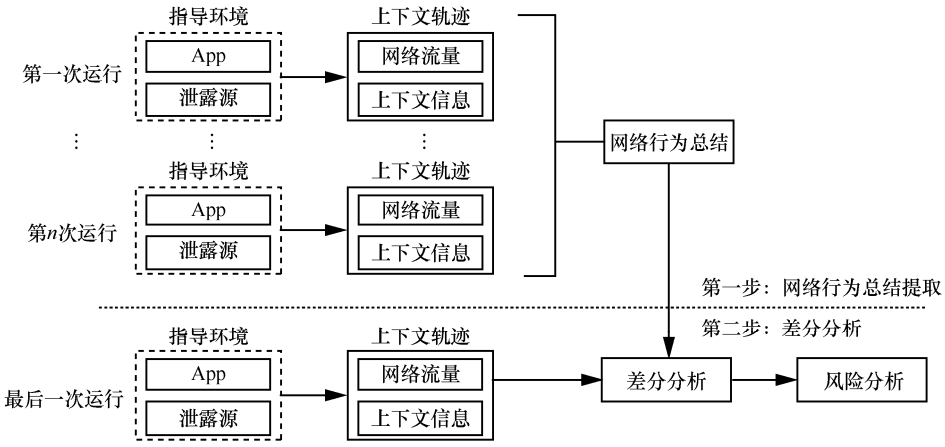


图 4　基于差分分析的方法流程

他类型的转换。与上述几种方法一样，该方法也不适用于通过 IPSec 加密的网络流量，不能对所有的 PII 泄露进行检测，因为使用 Monkey 方式采集数据，不能覆盖所有真实的用户输入操作。

2.4　基于静态数据污染的分析方法

文献[36]提出了一种将网络流量转换成服务（S, service）、键（K, key）、值（V, value）的三维数据集，然后使用静态污染提取算法进行 PII 探测的方法。定义服务使用如式(1)所示，其包含的值如式(2)所示，二者组成 domain-value 数据表。

$$\text{Domain} = \{\text{SK}_1, \text{SK}_2, \text{SK}_3, \cdots, \text{SK}_n\} \tag{1}$$

$$\text{Value} = \{v_1, v_2, v_3, \cdots, v_m\} \tag{2}$$

其中，n 为 SK（service key）的个数，m 为 SK 值的个数。静态污染提取算法将典型的 PII 标识为污染值类型作为输入值，使用信息流图自动且准确地跟踪 PII 类型及其位置。信息流图利用 domain-value 数据通过 2 个步骤的污染过程绘制而成，分别是域间感染（intra-domain infection）和域间路由（intra-domain routing）。域间感染方法基于以下原理：如果一个 SK 中的任何值是 PII，那么这个 SK 中的所有值都是 PII，从而在每个 SK 中选择相似的 PII。2 个 SK 之间由 2 个或多个共享值连接的关系称为域间路由，基于这些关系构建相邻矩阵，SK 组成节点集合 N={$\text{SK}_1, \text{SK}_2, \cdots, \text{SK}_n$}，节点关系集合为

$$\begin{aligned} E = \{&(\text{SK}_1, \text{SK}_2, v_1), (\text{SK}_1, \text{SK}_6, v_2), (\text{SK}_2, \text{SK}_3, v_3), \\ &(\text{SK}_2, \text{SK}_9, v_4), (\text{SK}_7, \text{SK}_9, v_5), (\text{SK}_3, \text{SK}_9, v_6), \\ &(\text{SK}_3, \text{SK}_4, v_7), (\text{SK}_4, \text{SK}_9, v_8), (\text{SK}_4, \text{SK}_7, v_9), \\ &(\text{SK}_4, \text{SK}_5, v_{10}), (\text{SK}_4, \text{SK}_8, v_{11}), (\text{SK}_5, \text{SK}_6, v_{12}), \\ &(\text{SK}_5, \text{SK}_8, v_{13}), (\text{SK}_6, \text{SK}_7, v_{14}), \cdots\} \end{aligned} \tag{3}$$

利用上述数据构造信息流图 IFG=(N,E)，如图 5 所示。该方法不需要在客户端部署任何其他应用程序就可以自动从大规模网络流量中提取 PII，也不需要任何手动工作就能对 ISP（Internet service provider）级网络流量进行反馈。

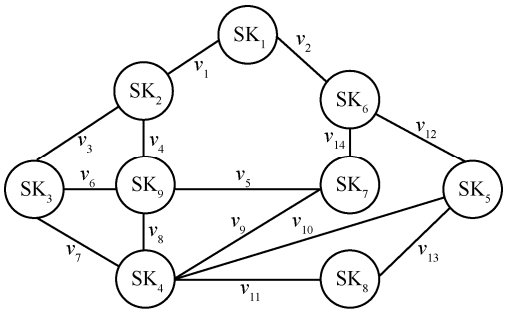


图 5　信息流图

3　in-App 操作信息探测

in-App 操作信息探测包括服务使用识别和用户动作识别。通过服务使用识别可以探测到用户在社交类 App 上发送的服务类型，例如在微信聊天界面发送图片、文字、语音电话等内容。但这种方法探测到的用户信息类型有限，往往用于辅助网络性能优化和用户使用习惯分析。用户动作识别则可以从加密流量中识别用户在某一特定 App 上进行了何种具体操作。与服务使用类型探测相比，这类信息探测更加细粒度。与个人标识信息探测方法的主动探测不同，in-App 操作完全被动地基于流量侧通道信息，其探测流程与网络流量分析流程几乎一样。网络流量分析流程如图 6 所示，通常包含 4 个步骤：数据采集、数据预处理、数据分析、结果评

估。特征选择是数据预处理中最核心的部分，数据分析则是各种用户信息探测模型的构建方法与理论。

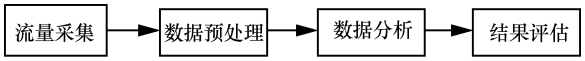


图 6 网络流量分析流程

3.1 数据采集

从层次化协议角度来看，可以从链路层、网络层及应用层采集数据，根据不同研究目的采取相应的数据采集方法，大致分为 Wi-Fi 接入点（AP, access point）、真实移动设备、Wi-Fi 监听、模拟器。通常采集数据时需要记录的信息有每个操作的起始时间、操作类型、App 名称。

1) Wi-Fi 接入点

移动设备链接 Wi-Fi 接入点，Wi-Fi 接入点再链接已经接入互联网的计算机，然后在计算机上使用流量采集软件进行数据采集。在进行大规模网络环境下的用户信息探测前，需要先过滤非移动设备的网络流量。通常使用与加密无关的数据链路层和网络层信息，以及 MAC 地址的组织唯一识别符（OUI, organization unique identifier）等信息作为流量过滤规则[37]。

2) 真实移动设备

最直接的数据采集方式是在移动设备上安装轻量级的流量记录程序，这种方式在对特定的单个或多个移动设备进行数据采集时非常高效。但安装这类数据记录 App 需要特定的权限。

3) Wi-Fi 监听

最简单的数据采集方法是将网卡设置成监听模式，被动监听附件的 Wi-Fi 信号[38]或使用专业的 Wi-Fi 信号监听器。这种方法有以下特性：① 多 AP 网络环境的数据采集涉及采集工具的多点部署、同步，以及从不同资源访问的时间问题。因此，直接在网关进行流量采集更准确和高效；② Wi-Fi 接入点存在高密集型的网络活动情景，在监听时对数据存储设备有一定的要求。

4) 模拟器

移动设备模拟器在运行时通过宿主机进行网络连接，因此可以将该宿主机作为网络流量的采集点。该类方式可以快速初始化多种手机品牌和操作系统版本的虚拟机，在 PC 端通过中间平台可以实现对其程序化控制，克服了直接在移动设备上采集数据时缺乏工具和资源受限这一困难。这使大规模 App 的网络流量分析非常高效快捷。

3.2 特征选择

在对加密流量进行分析前，根据信息探测的方法对输入数据进行格式转换、特征提取以及特征选择，本文将这些操作统称为特征选择。数据分组层面可直接使用的基本特征有数据分组大小、数据分组到达时间间隔、传输方向。

1) 数据分组序列

数据分组序列是按照数据分组产生的时间顺序组成的一组时间序列。每条时间序列本身没有固定的长度，但需要将其进行格式转换以适应特定的向量格式。文献[39]在特征提取阶段，根据 IP 和 TCP 头部中可利用的信息分离出目标 App 的网络流，将这些网络流与其他流用全局最小距离区分开，然后将网络流转换成时间序列，进而计算这些序列到最近代表点距离、产生这些序列动作的权重、序列的持续时间、序列内的数据分组数量和大小以及数据分组到达时间间隔，然后将上述特征值转换成特征向量。

2) 统计特征

根据数据分组到达时间间隔、传输方向对数据分组序列进行分组（burst 或 data block），然后选取主要特征（如数据分组大小、数据分组达到时间间隔、字节速率、上下行字节数之比等）的统计特征。例如，直方图通过将数据的分布情况用固定数量的箱子来表示，其中每个箱子对应一系列的值，表示这些值在这个范围发生的次数。通过运行统计测试（例如 t-test）可以评估数值的分布情况，从而选出具有代表性的特征。

3) 会话特征

通过学习不同的已知网络流量的协议会话流或数据分组的某些特征统计分布特性，形成分类器来识别网络流量，从会话层提取相关特征。但其实时性较差，需要先获取完整的会话数据流，然后统计数据分组个数、会话载荷平均长度等特征。这类方法选取的数据分组特征不受流量加密的影响。此外，移动设备的活动周期和通信机制一般都有预设的执行流程，会呈现一定的规律。因此，可以结合 App 的通信框架特征和外链特征构建有效特征，实现所需要的用户信息探测。该类方法只针对 SSL（secure socket layer）/TLS 模型识别有效，对 IPSec 加密无效。

4) 混合方法

有研究者使用深度学习方法进行网络流量的

分析。这类方法需要先对网络流量进行时间序列提取，然后才能输入深度学习算法中训练。文献[40]使用多层感知机（MLP, muti-layer perception）加一个隐藏层的方法，使用有效载荷的前 N 个字节、原始数据分组前 N 个字节、建立网络连接时的前 20 个数据分组的 6 类数据（源端口、目的端口、传输层载荷字节数、TCP 滑动窗口大小、数据分组到达时间间隔、数据分组传输方向）作为深度学习的输入数据。

3.3　模型与方法

本节从服务使用识别和用户动作识别两方面对 in-App 操作信息的研究进展进行阐述。

3.3.1　服务使用识别

由于移动设备操作系统自身的设计逻辑，用户在同一时刻只能操作一个 App。用户操作触发的网络流量传输往往需要一定时间，因此采集到的网络流量会存在单一服务使用和混和服务使用 2 种类型。这类研究存在以下几点特性。

1) 通常依赖于使用加密流量中包含的统计特征来识别用户的身份或行为。然而，用户移动设备使用模式的多样性使信息探测变得复杂。

2) 只对少数几款 App 中研究者认为用户隐私敏感的服务使用进行识别，包含发送图片、语音、视频电话、语音电话、定位信息等社交类 App 的聊天功能界面。

3) 利用专家知识对具体的 App 定制特征提取方法是这类研究的核心部分。结合机器学习算法进行服务使用类型识别，但是这部分通常不是该领域的研究重点。

服务使用识别主要有以下几种方法。

1) 统计特征与机器学习组合

文献[19]首先将在一定观察时间内（100 s）的流量通过有重叠的滑动窗口切分成片段（frame），然后对 frame 的到达时间、大小以及方向提取了 20 种统计特征，最后用机器学习算法来识别服务使用类别。该方法仅通过收集和分析少量的无线流量，就可以确定附近的每个移动设备用户是否使用了目标 App 以及进行了何种操作，得到随机森林算法最好的识别效果。这类方法中，不同研究工作的区别主要体现在特征选择方面。文献[41]首先在原始流量中提取了 30 种数据分组长度和时间的统计特征，利用 MIMD 准则选出 6 种最具代表性可迭代的流量特征，这样可以降低缓存要求并提高处理速度；然后利用连续时间窗口将进站流量切分成流量片段，采用 rCKC（recursive time continuity constrained k-means clustering）算法将来自同一服务使用类型的流量片段聚类在一起。文献[42]从行为结构、流量特征和时间依赖关系 3 个方面对流量进行建模，实现服务使用类型的分类。该方法首先用分层的方式将流量分割到会话中，而会话中又包含许多对话，这些对话大多是单一服务使用类型，对于混合服务使用类型的对话，则利用隐马尔可夫模型（HMM, hidden Markov model）检测并分解为单一服务使用类型的子对话；然后，提取数据分组长度和时延的序列特征，训练多个服务使用分类器。

2) 统计特征与哈希查表

文献[43]针对微信在关闭时不能接收传统网络流量的情况，通过识别和分析 iOS 推送通知服务 APN（access point name）流量，识别消息类型。首先，基于会话对流量进行集群，并将其划分为多个数据流块（brust）；然后，提取每个数据流块的统计特征，并将这些特征发送到分类器中，从背景流量中提取 APN 流量；最后，使用基于哈希的查找表方法分析来自 APN 流量的消息类型。这种方法能够推断的服务使用类型有限。

3) 自定义数据分组交互语言

文献[44]提出了一种基于 regex-inspired 语法的交互模式语言来描述 App 接收和发送数据分组长度序列的方法，实现对信息、语言和视频 3 种服务使用类型的规则构建，当一条包含已经定义的数据分组长度序列的网络流量到达识别模块时，根据规则匹配即可判定服务使用类别。该方法仅在 3 款主流的社交类 App 上验证了其识别性能。表 2 是该方法的示例，每条规则由表示数据分组长度序列的 item 序列组成，$\text{item}_i = x[\text{-}y][\{a,b\}]$，其中，$[\{a,b\}]$ 表示长度为 x 的数据分组重复出现的次数范围，$[\text{-}y]$ 表示其取值范围。例如当一条网络流量的数据分组长度序列是由长度为 3 B 和 52 B 重复组成时，则判定其为聊天信息。

表 2　自定义数据分组交互语言示例

行为	应用	规则
语音电话	微信	3{1,3}, 56-60{1,3}, 400-800
语音电话	微信	3{1,3}, 56-60{1,3}, 3{1,3}, 117 OR 3{1,3}, 56-60{1,3},3{1,3}, 114
聊天信息	微信	3{1,3}，52

4) 多标签多视图方法

之前的研究大多将流量切分成单一应用序列，然后利用机器学习算法将其分成不同的服务使用类型。这会造成属于同一服务使用类型的流量切分不准确的问题。对流量进行多标签多视图标注可以解决这一问题。多视图学习利用不同视角之间的一致性来探索数据的不同视图，以增强模型学习，达到更好的学习效果。多标签方案指每个示例都与多个类标签相关联，其结合了单一用途分类和混合用途分类，通过混合用途离群值的检测和处理，可以减少后续分类的步骤。

文献[45-46]通过增强的流量分割方法来减少混合使用的流量子序列；然后，利用流量子序列的数据分组长度视图和数据分组到达时间间隔视图的分类一致性，提高分类精度，如图 7 所示。此外，该方法还将单一使用子序列的分类问题和混合使用子序列的分类问题结合成一个统一的多标签逻辑分类问题，有助于输出每种服务使用类型的概率，在噪声环境下更加稳健。当序列分割方法不能清晰准确地分割序列时，会对该方法造成影响，从而产生混合多种用户动作或状态的子序列，进而影响服务使用类别的判定。

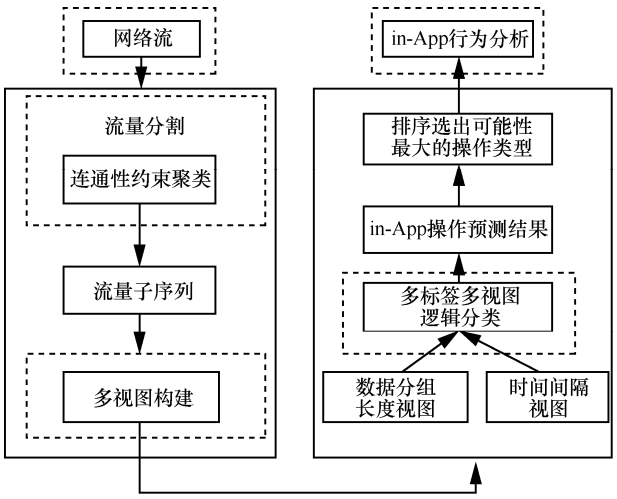


图 7 多标签多视图方法

3.3.2 用户动作识别

基于不同 App 用户交互设计模式的不同，特定的用户动作通常会产生区别性的网络流量轨迹模式。因此，这些模式可以用于从匿名网络中进行特定 App 用户动作识别。根据一段时间内用户动作模式的差异，可以从网络数据中标记属于特定用户的特定流量，也可以进行用户习惯分析。文献[13, 42, 47-52]的研究大都是针对隐私敏感性的即时通信类 App（如 iMessage、KakaoTalk、WhatsApp）、社交类 App（如 Facebook、Twitter）、邮件客户端类 App（如 Gmail、Yahoo Mail）进行用户动作识别，且大都针对某一款 App 的特定动作。

传输层的流量经过加密后，负载特征和流统计特征都发生了改变，因而不具备简单且易提取的特征字段。传统非加密流量识别方法包括基于深度分组检测、基于深度流检测、基于端口以及负载检测等方法，但都不适用于加密流量。加密和端口混淆等技术的使用，为加密网络协议和加密网络应用的细颗粒度识别和准确实时识别带来了挑战[53]。研究者使用机器学习算法来应对这些问题，在网络流量分析领域，这类算法通常需要基于 DTW（dynamic time warping）和 Jaccard 算法进行相似度计算，将相似网络流量聚类在一起，为后续的特征提取以及监督学习算法的使用建立有效标记的数据集。

用户动作识别主要有以下几种方法。

1) 字典模式

根据不同特征在不同网络行为下的变化程度这一重要特性，字典模式将具有稳定性和确定性的特征构建成基于哈希的查找表，并标记其对应的用户动作类别标签。当一个新的数据分组到达时，查询字典以检索用户动作所属的类别标签，如果返回一个标签，数据分组被给予那个标签；如果返回 2 个或多个标签，则选择训练期间最常与该特征关联的用户动作类别标签。文献[13]对 iOS 即时信息服务进行研究，揭示了目标用户和服务器之间交换数据分组大小的规律，发现了每个用户行为都有 2 种不同的数据分组大小：消息被发送到服务器时的数据分组大小、从服务器接收消息时的数据分组大小。该方法只对以下 5 种用户操作类型进行推断：开始打字、停止打字、发送文本、发送附件、阅读收信。此外，还可以推断用户使用的 6 种语言和交换信息的长度。该方法基于 2 个必要的假设：①对于用户动作识别，假设已经正确地推断出目标移动设备正在运行的操作系统类型；②对于语言和消息长度推断，假设已经正确识别了发送消息时的用户动作。

2) 统计特征与机器学习组合方法

统计特征与机器学习组合方法首先从数据分组序列中提取统计特征，然后利用机器学习算法识别用户操作类型。文献[54]基于此原理，针对微信提取双向流量中数据分组的总长度和总数量、数据

分组长度序列的 7 种统计特征以及数据分组长度分布特征，测试了 5 种常用的机器学习分类算法对细粒度的用户动作识别效果，发现只有随机森林算法取得了准确率和召回率均超过 92%的较好结果。文献[55]仅使用 7 种与数据分组到达时间间隔密切相关的统计特征，就实现了从一小部分流量中识别出细粒度的用户动作。对于 Facebook 上一篇文章的评论，该方法可以判断这个评论是长文字、短文字、贴纸还是图片。该方法只在 3 款 App 上验证了效果，且测试时没有考虑流量噪声的影响。

文献[56]首先基于时间阈值 1.25 s 将流量划分为几个描述不同操作的突发事件，然后从每个突发事件中提取相关的红包交易和资金转账特征：①数据分组长度的一阶和二阶统计特征；②数据分组长度在特定范围内的数据分组总数；③TCP 握手数量；④TCP 会话的出入流统计（总的有效载荷和数据分组数量），用来区分不同的突发事件。该方法可以准确识别红包交易和资金转账的行为，并准确预测红包交易和资金转账的数量。文献[57]用突发事件的流统计特征代替 TCP 握手数量，增加了对 App 的识别和用户动作细分识别。文献[58]基于上述 2 种方式在特征选择时进行明显简化，只选择了出站、进站、双向数据分组长度的统计特征，用于识别“比特币钱包”类 App 上用户共性动作，例如，启动 App、接收和发送比特币。

文献[59]首先通过 SNI（server name indication）从原始流量中提取出每个 App 的流量，然后根据文献[60]的方法从每条流中提取 22 种关于数据分组长度和时间的统计特征，最后用机器学习算法识别用户动作。在此基础上，对每个用户的 App 使用习惯提取特征（如 App 的类型、使用时间、使用频率、在 App 上操作最多的用户动作），构建分类器以识别用户的年龄、性别和职业类型。但该文献没有讲述其用户动作流的分割方法。

3) 基于凝聚层次聚类和随机森林算法

基于凝聚层次聚类和随机森林算法根据时间超时机制[10]将数据分组切分流量，变成数据分组序列，然后选取数据分组长度为作特征，用凝聚分层聚类算法将相近序列聚集在一起并选出类中心代表某一用户动作类型，再用随机森林算法推断新的数据分组序列中的用户动作类型。文献[12]利用该方法识别 Facebook、Twitter 和 Gmail 三款主流 App 的少数几种用户隐私敏感的用户动作类型。文献[48]扩展了另外 5 款 App 的用户动作识别。文献[47]基于此方法对一款即时聊天应用程序 Kakao Talk 的 11 种具体的用户行为（加入聊天室、发送信息、添加朋友等）进行识别。该方法只对 SSL/TLS 加密后的数据处理有效，不适于 IPSec 加密。

各类机器学习算法在用户信息探测中实际应用时，大多数情况下都需要结合多种算法才能完成相关网络流量识别和分析的任务。文献[48]结合监督学习和无监督学习识别用户在几款主流 App 上的用户动作指纹，首先将流进行分层聚类，然后将每个用户动作和每段流都分配到最近的类中，最后用随机森林算法训练分类器。

4) 流量行为模型

文献[49]基于用户在操作 App 时产生的流量行为（如传输速度、数据分组交换、数据移动）都是 App 开发者设置的这一认知，仅利用一段中间流量，首先将数据流切分成观察时间为 5 ms 的片段，提取发送和接收数据分组的平均到达时间间隔、分组计数比例、数据大小占比这 6 种特征和 20 种数据分组大小在 10 种区间内的数量；然后使用增量式 k-means 聚类算法找出最具有区分性的行为模型；最后通过多分类 SVM 将行为模型对应到用户动作。该方法能够仅基于检查 IP 标头的细微流量差异，就可对用户执行了何种动作进行推断。但该方法依赖于服务器名称解析，并要求网络工作流量与流量签名具有类似的时间顺序。由于传输连接的挑战，这种情况很难实现。

5) 深度学习

虽然传统的机器学习算法可以解决很多因流量加密而造成的问题，但仍然有一些不足。例如，耗时多，手工提取特征代价大，特征频繁更新。深度学习可以自动提取特征，近期引起很多研究者的重视，研究者开始尝试用深度学习的方式分析移动设备中加密流量的各类信息。文献[61]采用基于滑动窗口的方法将加密的流量流划分为一系列对应于单一 App 用户动作的流量片段，然后用归一化的时间−空间流量矩阵和流量谱向量表示各段流量，在归一化表示的基础上，开发了一种深度神经网络（DNN, deep neural network）分类算法，用于识别用户在不同 App 上进行的关键动作。文献[62]对数据分组负载首先利用卷积神经网络（CNN, convolutional neural network）提取空间依赖特征，然后用 LSTM 提取时间依赖特征，再提取 IP 头部信息和流

统计特征，并一起输入 DNN 模型中。该方法只对较粗粒度的用户操作进行识别。

6) 迁移学习

之前的大多数研究中，机器学习算法都是在相同设备和相同版本的 App 采集的数据集上进行训练和测试。文献[63]研究发现在未知设备和操作系统版本上测试已经训练好的模型，其识别的准确率均出现超过 21%的降低。因此，需要对每种不同移动设备和 App 版本的组合情况进行训练，但这非常不切实际。由于 App 版本、移动操作系统和设备模型（下文统称为配置）多种多样，以前的方法并不适用于实际场景中的配置，文献[64]将不同的配置作为迁移学习的案例，并采用协同训练的方法来支持迁移学习过程，利用了来自源配置的加密流量的少量标记实例，以便构造一个分类器，扩展了这些方法的能力，以泛化不同的配置，能够识别完全未标记的不同目标配置中的用户操作。

7) 基于最大熵原理

文献[65]针对 Instagram 使用请求、响应长度及其各自第一个 TLS 片段长度这 4 种特征来过滤特征稳定的 JSON（JavaScript object notation）格式数据；然后基于最大熵原理，提出了一种划分这些稳定特征分布范围的方法，并将特征空间映射到支持向量空间中，采用支持向量机算法进行用户动作识别。

3.4 评价体系

通常网络流量分析流程的最后一个阶段是结果评估，依照标准评估此前所采用的分析方法是否理想，常用的评价标准包括真正（TP, true positive）、假正（FP, false positive）、真负（TN, true negative）、假负（FN, false negative）、精确率（PR, precision rate）、召回率（RR, recall rate）及 $F1$ 值等。TP 表示将真正属于某类别的对象正确识定为该类别。FP 表示将不属于某类别的对象判定为该类别。TN 表示将不属于某类别的对象判定为非该类别。FN 表示将属于某类别的对象判定为非该类别。PR 表示正确判定为某一类别对象的占比，它能够反映识别的准确性。

$$\mathrm{PR}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{4}$$

RR 表示属于某一类别对象被正确判断的占比，它能够反映判断的全面性。

$$\mathrm{RR}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{5}$$

$F1$ 值是综合准确率和召回率的评价指标，$F1$ 值越高表示识别性能越好。

$$F1=\frac{2\mathrm{PR}\cdot\mathrm{RR}}{\mathrm{PR}+\mathrm{RR}} \tag{6}$$

4 未来研究方向和挑战

4.1 未知应用

未知应用指未知且流量识别器未经过训练的应用程序。在某些情况下，未知应用程序流量将占用大量的网络带宽。利用现有的技术，将能够识别的流量标记之后，如何将未标记过的流量进行同类聚合、预处理、打标签、训练，将未识别流量标记为某未知流量以供后续的标记和管理，也是从移动设备加密流量中探测用户信息时需要注意和克服的一个问题。此外，当前的研究大多只是在较少量的手机操作系统版本和手机型号上安装了几款主流 App 作为数据采集的平台，以这样的数据集训练的用户信息探测器的泛化性和稳健性往往不足，使其在不同场景下对新流量的分析能力降低。另外，App 版本的升级变化也会使用户信息探测的效果大打折扣。如果能及时自动更新用户信息探测器，则可以有效避免人为参与探测器的调试和更新。但目前还未发现有涉及该领域的相关研究。除了可以在正规应用市场下载 App 外，还有许多恶意 App 通过浏览器广告诱导用户下载安装，这类 App 对用户隐私探测和窃取更严重，因此也需要加强对这类 App 的用户信息泄露的防护。

4.2 对抗反检测

1) 代理技术

当前一些应用为了提高数据传输效率，采用数据压缩代理技术来减少数据规模，但这对从流量分析角度获取用户信息设置了障碍。通常这类数据压缩会使流量的统计特征发生较大的改变，使基于流统计特征的方法性能降低。为了克服各种类型的代理技术对用户信息探测的阻碍，可以考虑采用分层、多阶段的处理方式。

2) 流量伪装

流量伪装指将一种流量特征伪装成另一种流量特征，以此抵抗基于流量统计特征的流量分析。一些恶意应用常使用流量伪装方法进行网络攻击和隐蔽通信。为了应对加密流量中的异常与攻击，急需研究更加有效的流量伪装识别技术等其他的对抗反检测方法，如多路复用流技术。

3) 匿名通信

匿名通信指采取一定的措施隐蔽通信流中的通信关系，使窃听者难以获取或推断通信双方的关系及内容。匿名通信的目的就是隐蔽通信双方的身份或通信关系，保护网络用户的个人通信隐私。但这类技术具有很强的两面性，网络攻击者也会利用这些对抗反检测技术进行一些恶意的网络行为。

从层次化网络协议的角度来看，当前从流量分析角度对用户隐私泄露探测方面的研究是从网络层和传输层获取网络流量信息，但低层网络流量更适用于加密流量的分析。这方面的研究还相当缺乏，是未来可以关注的方向。

4.3 应对新协议

目前，不断有新型网络协议投入使用，例如 QUIC、SPDY、HTTP/2.0 等新协议，以及新版本的加密协议 TLS 1.3。此外，越来越多的 App 开发者使用私有协议，且这些协议的语法和语义并不规范。因此，之前基于旧版本协议的特征选择和模型建立方法可能不完全适用，需要进行周期性更新和改进，如何使其自适应地应用于新协议成为急需解决的重大问题，尤其很多用户信息探测方法基于已知协议的知识背景，无法识别未知协议流量给网络带来的安全问题，安全隐患巨大。

此外，考虑到移动设备与物联网（IoT, Internet of things）设备一样具有移动性和相近或完全相同的协议栈，因此可以尝试将移动设备网络流量分析的相关技术迁移到 IoT 设备的相关研究上。

4.4 负载均衡

之前的研究通过主要域名或 IP 地址来提取 in-App 操作的流量，例如利用 WHOIS 协议进行域过滤，然而这类启发式方法简单粗糙且容易出错。随着越来越多 App 的服务端部署在云服务器上，其采用的负载均衡策略使之不能将服务端 IP 地址与服务进行映射绑定。当前绝大多数 in-App 识别方法在对目标服务采集数据时刻意延长操作时间以获取足够的数据流量，方便提取有效特征。但这存在一个严重问题，即基于非用户正常操作逻辑的模式构建，难以直接应用在实际网络环境中。

5 结束语

对移动设备的加密流量分析是当前的研究热点，其中关于用户信息的探测也是其中重要的一个方面。本文结合近几年的相关研究成果，首先，归纳了关于加密流量分析与用户信息密切相关的研究目标。其次，阐述了用户信息探测的研究体系，其中包含数据采集、特征选择、模型与方法，以及评价体系，并对每个部分的研究状况进行了阐述和评论。最后，归纳现有基于移动设备加密流量用户信息探测的不足和难点，展望其研究趋势和未来的研究方向。

参考文献：

[1] 中国互联网络信息中心. 第 45 次《中国互联网络发展状况统计报告》[R]. (2020-04-28)[2020-07-14].
China Internet Network Information Center. The 45th "Statistical Report on Internet Development in China"[R]. (2020-04-28)[2020-07-14].

[2] ENCK W, GILBERT P, HAN S, et al. TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones[J]. ACM Transactions on Computer Systems, 2014, 32(2): 1-29.

[3] ARDAGNA C A, CONTI M, LEONE M, et al. An anonymous end-to-end communication protocol for mobile cloud environments[J]. IEEE Transactions on Services Computing, 2014, 7(3): 373-386.

[4] CONTI M, DRAGONI N, GOTTARDO S. MITHYS: mind the hand you shake-protecting mobile devices from SSL usage vulnerabilities[C]// International Workshop on Security and Trust Management. Berlin: Springer, 2013: 65-81.

[5] BERTHOLD O, FEDERRATH H, KOHNTOPP M. Project “anonymity and unobservability in the Internet”[C]//Proceedings of the Tenth Conference on Computers, Freedom and Privacy: Challenging the Assumptions. New York: ACM Press, 2000: 57-65.

[6] BUJLOW T, CARELA-ESPANOL V, BARLET-ROS P. Independent comparison of popular DPI tools for traffic classification[J]. Computer Networks, 2015, 76: 75-89.

[7] WRIGHT C V, COULL S E, MONROSE F. Traffic morphing: an efficient defense against statistical traffic analysis[C]//Proceedings of the 16th Network and Distributed Security Symposium. Piscataway: IEEE Press, 2009: 237-250.

[8] FAHL S, HARBACH M, MUDERS T, et al. Why eve and mallory love android: an analysis of android SSL (in)security categories and subject descriptors[C]//ACM Conference on Computer & Communications Security. New York: ACM Press, 2012: 50-61.

[9] GEORGIEV M, IYENGAR S, JANA S, et al. The most dangerous code in the world: validating SSL certificates in non-browser software[C]//ACM Conference on Computer & Communications Security. New York: ACM Press, 2012: 38-49.

[10] STOBER T, FRANK M, SCHMITT J B, et al. Who do you sync you are? Smartphone fingerprinting via application behaviour[C]// Proceedings of the sixth ACM conference on Security and privacy in wireless and mobile networks. New York: ACM Press, 2013: 7-12.

[11] RAYMOND J. Traffic analysis: protocols, attacks, design issues, and open problems[C]//Designing Privacy Enhancing Technologies. Berlin: Springer, 2001: 10-29.

[12] CONTI M, MANCINI L V, SPOLAOR R, et al. Can’t you hear me knocking: identification of user actions on android Apps via traffic analysis[C]//Proceedings of the 5th ACM Conference on Data and

Application Security and Privacy. New York: ACM Press, 2015: 297-304.

[13] COULL S E, DYER K P. Traffic analysis of encrypted messaging services: Apple iMessage and beyond[J]. ACM Sigcomm Computer Communication Review, 2014, 44(5): 6-11.

[14] LI H X, ZHU H J, MA D. Demographic information inference through meta-data analysis of Wi-Fi traffic[J]. IEEE Transactions on Mobile Computing, 2018, 17(5): 1033-1047.

[15] WANG P, CHEN X, YE F, et al. A survey of techniques for mobile service encrypted traffic classification using deep learning[J]. IEEE Access, 2019, 7: 54024-54033.

[16] WANG X, QIN X, HOSSEINI M B, et al. GUILeak: tracing privacy policy claims on user input data for Android applications[C]// International Conference on Software Engineering. Piscataway: IEEE Press, 2018: 37-47.

[17] CONTI M, LI Q Q, MARAGNO A, et al. The dark side(-channel) of mobile devices: a survey on network traffic analysis[J]. IEEE Communications Surveys and Tutorials, 2018, 20(4): 2658-2713.

[18] NABOULSI D, FIORE M, RIBOT S, et al. Large-scale mobile traffic analysis: a survey[J]. IEEE Communications Surveys and Tutorials, 2016, 18(1): 124-161.

[19] WANG Q, YAHYAVI A, KEMME B, et al. I know what you did on your smartphone: inferring App usage over encrypted data traffic[C]//2015 IEEE Conference on Communications and Network Security. Piscataway: IEEE Press, 2015: 433-441.

[20] JIA Q, ZHOU L, LI H, et al. Who leaks my privacy: towards automatic and association detection with GDPR compliance[C]//International Conference on Wireless Algorithms, Systems, and Applications. Berlin: Springer, 2019: 137-148.

[21] LE A, VARMARKEN J, LANGHOFF S, et al. AntMonitor: a system for monitoring from mobile devices[C]//Proceedings of the 2015 ACM SIGCOMM Workshop on Crowdsourcing and Crowdsharing of Big (Internet) Data. New York: ACM Press, 2015: 15-20.

[22] RAZAGHPANAH A, VALLINA-RODRIGUEZ N, SUNDARESAN S, et al. Haystack: in situ mobile traffic analysis in user space[J]. arXiv Preprint, arXiv: 1510.01419, 2015.

[23] NOVAK E, AUNG P T, DO T. VPN+ towards detection and remediation of information leakage on smartphones[C]//2020 21st IEEE International Conference on Mobile Data Management. Piscataway: IEEE Press, 2020: 39-48.

[24] SONG Y, HENGARTNER U. PrivacyGuard: a VPN-based platform to detect information leakage on android devices[C]//Proceedings of the 5th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices. New York: ACM Press, 2015: 15-26.

[25] SHUBA A, LE A, ALIMPERTIS E, et al. AntMonitor: a system for on-device mobile network monitoring and its applications[J]. arXiv Preprint, arXiv: 1611.04268, 2016.

[26] BAKOPOULOU E, SHUBA A, MARKOPOULOU A J A P A. Exposures exposed: a measurement and user study to assess mobile data privacy in context[J]. arXiv Preprint, arXiv:2008.08973, 2020.

[27] SRIVASTAVA G, BHUWALKA K, SAHOO S K, et al. Privacyproxy: leveraging crowdsourcing and in situ traffic analysis to detect and mitigate information leakage[J]. arXiv Preprint, arXiv: 1708.06384, 2017.

[28] REN J, RAO A, LINDORFER M, et al. ReCon: revealing and controlling PII leaks in mobile network traffic[C]//Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services. New York: ACM Press, 2016: 361-374.

[29] RAO A, KAKHKI A M, RAZAGHPANAH A, et al. Using the middle to meddle with mobile[R]. (2013-12-10)[2020-07-14].

[30] SHUBA A, BAKOPOULOU E, MEHRABADI M A, et al. Antshield: on-device detection of personal information exposure[J]. arXiv Preprint, arXiv: 1803.01261, 2018.

[31] SHUBA A, BAKOPOULOU E, MARKOPOULOU A. Privacy leak classification on mobile devices[C]//2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications. Piscataway: IEEE Press, 2018: 1-5.

[32] JIN H, LIU M, DODHIA K, et al. Why are they collecting my data? Inferring the purposes of network traffic in mobile Apps[J]. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2018, 2(4): 1-27.

[33] TAN H Z, ZHAO W, SHEN H H. A context-perceptual privacy protection approach on Android devices[C]//2018 IEEE International Conference on Communications. Piscataway: IEEE Press, 2018: 1-7.

[34] JUNG J, SHETH A, GREENSTEIN B, et al. Privacy oracle: a system for finding application leaks with black box differential testing[C]//Proceedings of the 15th ACM Conference on Computer and Communications Security. New York: ACM Press, 2008: 279-288.

[35] CONTINELLA A, FRATANTONIO Y, LINDORFER M, et al. Obfuscation-resilient privacy leak detection for mobile Apps through differential analysis[C]//Network and Distributed System Security Symposium. Virginia: the Internet Society, 2017: 1-15.

[36] LIU Y, LIAO L, SONG T J S C I S. Static tainting extraction approach based on information flow graph for personally identifiable information[J]. Science China Information Sciences, 2020, 63(3): 1-17.

[37] ZHANG F, HE W, CHEN Y, et al. Thwarting Wi-Fi side-channel analysis through traffic demultiplexing[J]. IEEE Transactions on Wireless Communications, 2014, 13(1): 86-98.

[38] ATKINSON J S, MITCHELL J E, RIO M, et al. Your Wi-Fi is leaking: what do your mobile Apps gossip about you?[J]. Future Generation Computer Systems-the International Journal of Escience, 2018, 80: 546-557.

[39] CHENG Z, CHEN X, ZHANG Y, et al. Detecting information theft based on mobile network flows for android users[C]//2017 International Conference on Networking, Architecture, and Storage (NAS). Piscataway: IEEE Press, 2017: 1-10.

[40] ACETO G, CIUONZO D, MONTIERI A, et al. Mobile encrypted traffic classification using deep learning: experimental evaluation, lessons learned, and challenges[J]. IEEE Transactions on Network and Service Management, 2019, 16(2): 445-458.

[41] LIU J, FU Y, MING J, et al. Effective and real-time in-App activity analysis in encrypted internet traffic streams[C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2017: 335-344.

[42] FU Y, XIONG H, LU X, et al. Service usage classification with encrypted internet traffic in mobile messaging Apps[J]. IEEE Transactions on Mobile Computing, 2016, 15(11): 2851-2864.

[43] ZHANG Q, XU M, ZHENG N, et al. Identifying WeChat message types without using traditional traffic[J]. Information, 2020, 11(1): 18.

[44] PAPADOGIANNAKI E, HALEVIDIS C, AKRITIDIS P, et al. OTTer:

a scalable high-resolution encrypted traffic identification engine[C]// International Symposium on Research in Attacks, Intrusions, and Defenses. Berlin: Springer, 2018: 315-334.

[45] FU Y, LIU J, LI X, et al. Service usage analysis in mobile messaging apps: a multi-label multi-view perspective[C]//2016 IEEE 16th International Conference on Data Mining. Piscataway: IEEE Press, 2016: 877-882.

[46] FU Y J, LIU J M, LI X L, et al. A multi-label multi-view learning framework for in-App service usage analysis[J]. ACM Transactions on Intelligent Systems and Technology, 2018, 9(4): 40.

[47] PARK K, KIM H. Encryption is not enough: inferring user activities on Kakaotalk with traffic analysis[C]//Workshop on Information Security Applications. Berlin: Springer, 2015: 254-265.

[48] CONTI M, MANCINI L V, SPOLAOR R, et al. Analyzing android encrypted network traffic to identify user actions[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(1): 114-125.

[49] SALTAFORMAGGIO B, CHOI H, JOHNSON K, et al. Eavesdropping on fine-grained user activities within smartphone Apps over encrypted network traffic[C]//10th USENIX Workshop on Offensive Technologies. Berkeley: USENIX Association, 2016: 69-78.

[50] SAPIO A, LIAO Y, BALDI M, et al. Per-user policy enforcement on mobile Apps through network functions virtualization[C]//Proceedings of the 9th ACM workshop on Mobility in the evolving internet architecture. New York: ACM Press, 2014: 37-42.

[51] TONGAONKAR A, DAI S, NUCCI A, et al. Understanding mobile App usage patterns using in-App advertisements[C]//International Conference on Passive and Active Network Measurement. Berlin: Springer, 2013: 63-72.

[52] LI K, LI H, ZHU H S, et al. Side-channel information leakage of traffic data in instant messaging[C]//2019 IEEE 38th International Performance Computing and Communications Conference. Piscataway: IEEE Press, 2019: doi.org/ 10.1109/ IPCCC47392. 2019. 8958775.

[53] DYER K P, COULL S E, RISTENPART T, et al. Peek-a-Boo, I still see you: why efficient traffic analysis countermeasures fail[C]//2012 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2012: 332-346.

[54] HOU C, SHI J, KANG C, et al. Classifying user activities in the encrypted WeChat traffic[C]//2018 IEEE 37th International Performance Computing and Communications Conference. Piscataway: IEEE Press, 2018: 1-8.

[55] PATHMAPERUMA M H, RAHULAMATHAVAN Y, DOGAN S, et al. In-App activity recognition from Wi-Fi encrypted traffic[C]//Science and Information Conference. Berlin: Springer, 2020: 685-697.

[56] YAN F, XU M, QIAO T, et al. Identifying WeChat red packets and fund transfers via analyzing encrypted network traffic[C]//2018 17th IEEE International Conference On Trust, Security and Privacy in Computing and Communications/ 12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE). Piscataway: IEEE Press, 2018: 1426-1432.

[57] WANG Y, ZHENG N, XU M, et al. Hierarchical identifier: application to user privacy eavesdropping on mobile payment App[J]. Sensors (Basel), 2019: doi. org/10.3390/s19143052.

[58] AIOLLI F, CONTI M, GANGWAL A, et al. Mind your wallet's privacy: identifying Bitcoin wallet Apps and user's actions through network traffic analysis[C]//Proceedings of the 34th ACM/SIGApp Symposium on Applied Computing. New York: ACM Press, 2019: 1484-1491.

[59] NAIK M, BHATIA A, TIWARI K. I know who you are: a learning framework to profile smartphone users[C]//2020 International Conference on Communication Systems & Networks. Piscataway: IEEE Press, 2020: 555-558.

[60] MOORE A, ZUEV D, CROGAN M. Discriminators for use in flow-based classification[R] . (2005-08)[2020-07-14].

[61] LI D, LI W, WANG X, et al. ActiveTracker: uncovering the trajectory of App activities over encrypted internet traffic streams[C]//2019 16th Annual IEEE International Conference on Sensing, Communication, and Networking. Piscataway: IEEE Press, 2019: 1-9.

[62] HOU T, WANG T, LU Z, et al. Smart spying via deep learning: inferring your activities from encrypted wireless traffic[C]//IEEE Global Conference on Signal and Information Processing. Piscataway: IEEE Press, 2019: doi.org/ 10.1109/GlobalSIP45357.2019.8969428.

[63] ALAN H F, KAUR J. Can Android applications be identified using only TCP/IP headers of their launch time traffic?[C]// ACM Conference on Security & Privacy in Wireless & Mobile Networks. New York: ACM Press, 2016: 61-66.

[64] GROLMAN E, FINKELSHTEIN A, PUZIS R, et al. Transfer learning for user action identication in mobile Apps via encrypted trafc analysis[J]. IEEE Intelligent Systems, 2018, 33(2): 40-53.

[65] WU H, WU Q, CHENG G, et al. Instagram user behavior identification based on multidimensional features[C]//IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops. Piscataway: IEEE Press, 2020: 1111-1116.

[作者简介]

张腾飞（1991− ），男，河南洛阳人，中山大学博士生，主要研究方向为网络安全和网络行为分析。

余顺争（1958− ），男，江西景德镇人，博士，中山大学教授、博士生导师，主要研究方向为隐半马尔可夫模型和机器学习、协议逆向工程、网络流量和网络行为分析、异常检测、基于 SDN 的云安全。