Apps, Trackers, Privacy, and Regulators
A Global Study of the Mobile Tracking Ecosystem

Abbas Razaghpanah * Rishab Nithyanand † Narseo Vallina-Rodriguez ‡
Srikanth Sundaresan § Mark Allman ¶ Christian Kreibich ‖ Phillipa Gill **

* Stony Brook University
† Data & Society Research Institute
‡ IMDEA Networks and ICSI
§ Princeton University
¶ ICSI
‖ Corelight and ICSI
** University of Massachusetts Amherst

***Abstract*—Third-party services form an integral part of the mobile ecosystem: they ease application development and enable features such as analytics, social network integration, and app monetization through ads. However, aided by the general opacity of mobile systems, such services are also largely invisible to users. This has negative consequences for user privacy as third-party services can potentially track users without their consent, even across multiple applications. Using real-world mobile traffic data gathered by the Lumen Privacy Monitor (Lumen), a privacy-enhancing app with the ability to analyze network traffic on mobile devices in user space, we present insights into the mobile advertising and tracking ecosystem and its stakeholders. In this study, we develop automated methods to detect third-party advertising and tracking services at the traffic level. Using this technique we identify 2,121 such services, of which 233 were previously unknown to other popular advertising and tracking blacklists. We then uncover the business relationships between the providers of these services and characterize them by their prevalence in the mobile and Web ecosystem. Our analysis of the privacy policies of the largest advertising and tracking service providers shows that sharing harvested data with subsidiaries and third-party affiliates is the norm. Finally, we seek to identify the services likely to be most impacted by privacy regulations such as the European General Data Protection Regulation (GDPR) and ePrivacy directives.**

I. Introduction

Mobile applications have become increasingly central to our daily lives, providing us with a variety of services and utilities (often at no cost). We entrust apps with a wealth of information that enables them to carry out these functions, yet despite our reliance on these apps and our countless daily interactions with them, we know very little about what they share about us with third-parties, who these third-parties are, and what they do with our data. As with the Web [1], many mobile app developers integrate third-party services in their apps for a variety of purposes including app maintenance (*i.e.*, crash reports), analytics services, user engagement, A/B testing, social network integration, and advertising. Third-party services inherit the set of application permissions requested by the host app, allowing them access to a wealth of valuable user data, often beyond what they need to provide the expected service to the app developer or the end-user [2]–[4], particularly if the same library is used by multiple apps with different permissions.

This has direct consequences for user privacy. However, most third-party services, with the exception of online advertising services, operate in the background and *do not provide any visual clues inside the apps*, effectively tracking users without their knowledge or consent while remaining virtually invisible. The general lack of transparency in mobile systems leaves users unable to identify the third-party services used by their apps, let alone know to which extent these services are able to collect, correlate, and aggregate their personal data and online activity across apps and platforms. As a result, end users and developers alike have no insight into how these services operate at the network level, whether or how they handle sensitive data, and once the data leaves the device, whether they further share (or sell) it with other third parties, including affiliated advertising services and even data brokers. Despite the enormous research efforts conducted by academics and regulators to illuminate this ecosystem [5]–[9], there is a dearth of knowledge when it comes to understanding at scale the companies that own these services, where they operate, their relationships and partnerships, and what their privacy and data sharing policies are.

In this work, we focus on studying third-party services whose main function relies on collecting tracking information from users, which we henceforth refer to as Advertising and Tracking Services (**ATS**). To understand how mobile apps that utilize ATSes operate under the hood, and the privacy cost of using them for the users, we need a holistic view of the complex ecosystem of mobile ATSes. However, gathering large-scale comprehensive data from mobile apps is challenging [3], [10], [11]. Use of large-scale traffic traces gathered from ISPs lacks contextual information that is only available on the device (*e.g.*, flow-to-app mapping) [12]; and while relying on dynamic analysis to study apps [4] yields good coverage of tracking activities and can scale up via use of "UI monkeys" to synthesize user input [13]–[16], it lacks the depth achieved when analyzing apps using real user input, especially when apps require users to log in [17]. Similarly, use of static analysis [18]–[20] has allowed studying mobile applications without running or interacting with them, but requires a significant amount of manual inspection and validation, thus making it difficult to scale.

To overcome the limitations of existing mobile app analysis methods, we leverage Lumen Privacy Monitor (Lumen) [1], a mobile app that provides both users and researchers insight into network traffic generated by all apps, from the vantage point

[1] https://www.haystack.mobi

Network and Distributed Systems Security (NDSS) Symposium 2018
18-21 February 2018, San Diego, CA, USA
ISBN 1-1891562-49-5
http://dx.doi.org/10.14722/ndss.2018.23353
www.ndss-symposium.org

of the device itself and with real user-stimuli while operating entirely in user-space and without requiring root access. Lumen is available for free on Google Play, and provides us with anonymized, yet rich app traffic data from its users.

We analyze data gathered by Lumen and other publicly-available resources to make the following contributions:

Identifying mobile ATSes. We develop an automated approach to identify third-party services, ATSes, and any other ATS-capable services whose primary service does not appear to be ATS-related, yet perform harvesting of unique identifiers (**ATS-C**, Section IV). With our approach we are able to identify 2,121 ATS domains of which 233 were previously unreported in popular commercial ATS lists and 730 ATS-C services.

Uncovering parent companies. We then perform a characterization of the 2,121 ATS domains identified by our approach (Section V). We obtain the parent organizations of these services (after accounting for business mergers and acquisitions) and identify the dominant organizations in the mobile ATS ecosystem. We find that Alphabet-owned ATSes have presence in over 73% of apps in our dataset. This raises questions about Alphabet's monopoly in the mobile ATS ecosystem.

Cross-device tracking. Advertising and tracking services seek new mechanisms to track users across devices and platforms (*i.e.*, cross-device tracking) [22], [23]. We find that there is a high proliferation of cross-device tracking services, with 39% of our ATS domains. 17 of the top-20 largest ATS organizations have a presence both on the Web and in the mobile ecosystem.

Privacy policies and user transparency. Our analysis of the privacy policies of the most dominant ATS organizations (having presence in over 80% of all apps in our study) reveals that eight of the top-10 organizations reserve the right to sell or share data with other organizations, while all of them reserve the right to share data with their subsidiaries. This demonstrates that a small number of companies have a monopoly on controlling a large portion of the ecosystem and that they have the ability to track users and share the tracking data with other entities, all with little to no transparency.

Understanding the global flow of data and the implications of current and proposed regulations. We study how ATS-related data is exported across borders and explore the impact of the European General Data Protection Regulation (GDPR) and ePrivacy directives on ATSes (Section VI). We find that the European regulations are likely to impact ATS providers in the United States and China more significantly than others.

II. The Mobile Advertising and Tracking Ecosystem

A study of the privacy policies of some of the well-known dominant players in the third-party mobile advertising and tracking ecosystem reveals their lax data sharing policies and demonstrates that the flow of user private information can violate the user-app trust relationships when it is collected by third parties. Some of these services explicitly claim that they do not sell their data to third-parties, but their parent companies

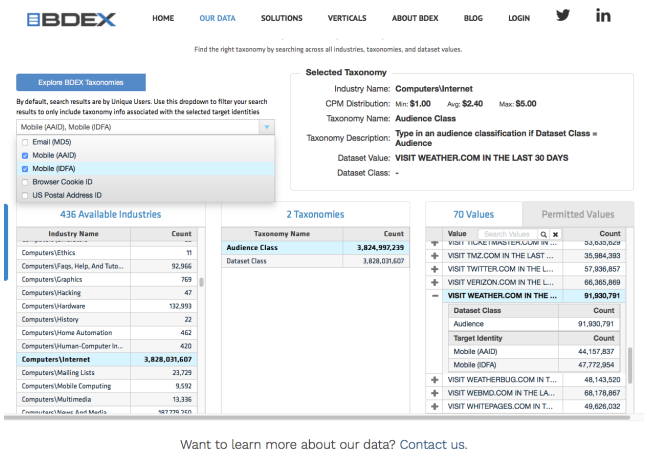

Fig. 1: A screenshot of a data exchange website allowing potential customers to query sample data.

allow data sharing between subsidiaries in their privacy policy–*e.g.*, Facebook Graph API can share its data with Facebook Ads. Others, explicitly say that they reserve the right to share aggregate (or sometimes even non-anonymized) data with their, often undisclosed, third-party "partners". This is not helped by the fact that there have been numerous cases where companies with large amounts of user data have been found to sell this data to other companies [24].

Although some users may become indirectly aware of this data sharing between tracking and advertising companies when they start seeing related targeted ads in seemingly unrelated apps and websites [25], the data (anonymized or otherwise) mostly ends up in the hands of data brokers and exchanges where it can be sold to the highest bidder without their knowledge. Figure 1 shows a screenshot of one such data exchange's website, demonstrating the volume of rich data they claim to possess from mobile users alone. This is also significant because data breaches are a common occurrence that have affected even big players in the mobile world, with notable cases like Yahoo (Flurry's parent company) suffering from massive data breaches in 2014 and 2016 [26], [27] that compromised private information of hundreds of millions of users.

Mobile tracking and regulatory agencies. Regulatory agencies and policy makers have developed several laws to protect user privacy with relative success. However, each jurisdiction has its own rules and regulations regarding data collection to protect citizens against unlawful and invasive online tracking practices. One example of these regulations is the Children's Online Privacy Protection Act (COPPA) rule in the United States which bans the collection of private information from children and minors under the age of 13 without parental consent [28]. Despite this, there have been cases in the past where companies have been found in violation of COPPA and fined by the Federal Trade Commision (FTC) [29], and there are still countless examples of games and children's apps that use third party services collecting tracking data without parental consent [30]. The European Union has also proposed a regulation to enforce data protection for individuals

in the European Union. Named the General Data Protection Regulation (GDPR) [31], it will impose strict rules on tracking and collection of personal data within the European Union when it goes into effect. With app markets containing millions of apps, and given the lack of knowledge about the mobile tracking ecosystem, it is difficult to enforce these regulations at scale.

Analyzing the mobile ecosystem at a global scale. Previous work has shown that 99% app traffic is sent over the network, with only 1% of app traffic going over other channels like SMS [32]. This means that in order to study interactions between apps and third-party advertising and tracking services we need to analyze apps at the traffic level. Previous techniques to study these interactions trade off scale and comprehensiveness, either missing valuable contextual information about flows (*e.g.*, flow-to-app attribution and encrypted traffic) in network-level traffic analysis, or lacking scale and real-world user stimuli in static analysis. To avoid these pitfalls, we need a scalable measurement platform to study app behavior at scale and with access to rich on-device information.

In order to study these services, we need to identify the set of domain names reached by mobile application and classify them by their purpose. However, studying and classifying domains is still an open research challenge: it is therefore difficult to find out what role each domain plays in an app, not to mention attributing ownership.

First-party domains enable functionalities that are central to the app itself —although they can still track users— while third-party domains enable third-party functions in other apps. For example, Facebook domains are a central part of the Facebook app, while the same domains are considered third parties when they are used in other apps (*e.g.*, to provide Facebook API integration [33]). Our focus is on third-party domains since first-party domains are considered to be trusted by users when they install apps to provide a function, while third-party tracking domains may not. Moreover, third-party domains can still collect tracking information even when they do not provide in-app advertisements. Unfortunately, existing resources to identify third-party advertising and tracking domains often focus on the desktop platform, leaving out a vast number of domain names that are active only in the mobile space.

We define two categories of third-party domains based on their business model and observed behavior:

- **ATS domains** are ones that belong to companies whose primary service is providing advertising and tracking services, either to display targeted advertisements or for analytics and other tracking-focused purposes.
- **ATS-capable domains (ATS-C)** are domains that collect tracking information, but whose primary service is not specifically providing ads and analytics to app developers.

An example of an ATS service would be analytics services that monitor and report user information, device information, in-app events, and other events; while an ATS-C service like an integrated map API might collect location data and other information to provide area maps and directions to the app, but doesn't necessarily rely on tracking users for monetizing their service. However, it should be noted that this doesn't mean that ATS-Cs can't later share their data with third parties or sister companies and subsidiaries.

UID	Description	UID	Description
IMEI (✓)	Device ID.	**AndId** (✓)	Advertising ID.
IMSI (✓)	SIM ID.	**Phone #** (✓)	Phone number.
SIM# (✓)	SIM number.	**Fingerprint**	Device ID.
AndSerial	OS ID.	**MAC**	Unique hardware ID.

TABLE I: List of UIDs monitored by Lumen and their consent requirements. A ✓ indicates that consent of the user is required before harvesting the corresponding UID.

III. Lumen Privacy Monitor

Lumen Privacy Monitor is a home-built Android app that aims to promote mobile transparency and user awareness by informing users about how their installed apps handle sensitive data such as unique identifiers and personally identifiable information (PII). Table I lists the UIDs monitored and their consent requirements. It runs locally on the device and intercepts all network traffic—both over WiFi and the mobile network–without requiring root permissions. Lumen has been publicly available on Google Play since October 2015. We use anonymized traffic logs from over 11,000 crowdsourced Lumen (Lumen)[2] users for this study.

A. Lumen overview

Lumen works by leveraging the Android VPN permission to capture and analyze network traffic, including encrypted flows, locally on the device and in user-space. Lumen inserts itself as a middleware between apps and the network interface.

The use of the VPN permission to analyze app traffic on user-space is not novel [34], [35]. However, previous tools features essential to comprehensively study mobile traffic without affecting app execution, including efficient flow-reassembly (the ability to parse entire flows) and a non-distruptive TLS interception module. For further details about Lumen's goals, design, architecture, capabilities and ethical considerations we refer the reader to our previous report [21] and the project website [3].

B. Traffic analysis

By operating locally on the device, Lumen is able to correlate disparate and rich contextual information such as process IDs with flows. Lumen uses this vantage point and deep packet inspection techniques to analyze app payload and identify personal and sensitive data exported by apps. Beyond extracting traffic from regular flows, Lumen also deflates compressed flows and identifies privacy leaks obfuscated using different encoding mechanisms. To a large extent, the features implemented by Lumen can be perceived as a mobile and user-centric conception of the technologies present in Intrusion Detection Systems [36]–[38].

Additionally, since a large fraction of mobile apps have adopted TLS as the default protocol for data communications,

[2]The tool was initially called Haystack.
[3]https://www.haystack.mobi

Lumen employs a transparent man-in-the-middle (MITM) proxy for TLS traffic with user consent. At install time, Lumen explains the purpose of performing TLS interception and requests the user for permission to install a self-signed CA certificate in the root certificate store.

C. User privacy considerations

Examining user traffic, especially encrypted flows, raises privacy concerns. Lumen aims to preserve user privacy while gathering data regarding the export of unique identifiers and other PII. The app was built in consultation with our university IRB which deemed the current framework as not involving human subjects since its focus is on analyzing the behavior of software, not its users. In spite of this, we follow the principles of informed consent and additionally require users to opt-in twice before initiating traffic interception [39].

Lumen preserves user privacy by performing flow processing and analysis on the device, only sending summarized and anonymized data (Section III-D) for research purposes. We emphasize that Lumen does not send back any unique identifiers, device fingerprints, or raw captures. To further protect user privacy, the Lumen app also: 1) ignores all flows generated by browser apps (which may potentially deanonymize a user); and 2) allows the user to disable traffic interception at any time.

D. Lumen Data Summary

At the time of this study (August 2017), the Lumen dataset included the ports, origin app, destination domain, requested app permissions, and IP address, TLS-handshake information, and types of unique identifiers leaked of over 8.5M flows from 14,599 apps to 40,553 unique fully-qualified domain names (FQDNs) with 13,453 unique second-level domains (SLDs). For devices carrying SIM cards, we also associate the Mobile Country Code (**MCC**) with each flow. This data was collected from more than 11,384 Lumen users. Although 73% of all Lumen installs are from USA, Spain, Italy, Germany, and India, our dataset includes users from over 100 different countries[4] and is therefore able to provide insights into the pervasiveness of mobile ATSes globally.

We download metadata for each app in our dataset from Google Play and find that of the 14,599 monitored by Lumen, 3.0 % are paid apps, 26 % are free and allow in-app purchases[5], 16 % are not listed on Google Play, and the remaining apps are free. The set of apps not available on Google Play range from basic Android services to alternative app stores (*e.g.*, MoboGenie [40] and Aptoide [41]), removed apps (*e.g.*, Free WhatsDog [42]), and pre-installed apps from a number of mobile OS vendors (*e.g.*, HTC, LG, Samsung, and Cyanogen). Our monitored apps fall under 33 different Google Play categories, namely Games (21.8%)[6], Educational apps (9.9%), and Tools (7.9%). Finally, we consider the apps in our dataset to be representative of those used by average mobile users. 48 % of the apps in our records have more than 1M installs while 71 % of our measured apps are listed on the Google Play Top-50 charts for USA, Spain, Germany, India and UK.

[4]Statistics provided by the Google Play app developer console.

[5]Free apps can require in-app payments to enable a feature in the app (*e.g.*, to unlock new stages on an arcade game).

[6]The Games category groups together 14 different game subcategories such as Arcade, Puzzle, and Adventure.

IV. Domain Classification Approach

To identify ATS-related domains, we first need to distinguish third-party destinations contacted by apps from first-party ones, and then ATS-related domains from those associated with other third parties such as audio/video streaming SDKs and content delivery networks (CDNs). To filter out first-party domains, we analyze the domain names of destinations as they give us clues about who owns and operates them. We describe the problems with existing ATS blacklists and our approach for identifying third-party domains and ATSes.

A. Web-based ATS blacklists and URL categorization services

Existing ATS blacklists such as EasyList [43] and MalwareBytes hpHosts [44], and URL categorization services such as the McAfee URL Classifier [45], OpenDNS Domain Tagger [46], and VirusTotal [47][7]. can help us in our effort to identify and categorize mobile third-party services, we cannot rely on them completely for several reasons:

- Existing blacklists are specifically directed at identifying Web-specific ATSes and therefore do not account for many ATSes that operate exclusively in the mobile ecosystem.
- URL categorization services operate at the SLD level and not by the FQDN. As a result, they cannot accurately categorize subdomains used for purposes other than that of the original SLD. One such example is `graph.facebook.com`, a subdomain used by the Facebook Graph API (a known ATS), categorized as social networking due to `facebook.com` being categorized as such; or ATSes that are subdomains of known CDNs being mis-categorized as CDN.
- Even when considering only SLDs, the insight provided by URL classifiers can be incomplete as they are manually populated. Table II shows that while over 85% of the 13,453 Lumen observed SLDs have categories assigned to them by both McAfee and VirusTotal, OpenDNS has below 38% coverage. We see that 2.9 % of the SLDs in our dataset are unclassified in all three services.
- Categories returned by these services may be vague and not descriptive of the services provided by the domain; *e.g.*, `crashlytics.com`, a provider of bug tracking and analytics services, is categorized as "Software/Hardware" by the McAfee service.

Therefore, rather than completely relying on these lists, we use them to train, test, and curate our domain classifier and its results.

B. Our classification approach

Using Lumen data, we classify domains contacted by apps to distinguish third-party domains from first-party ones and then automatically categorize third-party services to identify ATS and ATS-C domains, as described below:

[7]VirusTotal aggregates the insight provided by URL classifiers such as Websense ThreatSeeker and Dr. Web.

	McAfee	OpenDNS	VirusTotal
SLD Coverage (%) [N=13,453]	96	37	85
Total / ATS-related categories	89 / 6	59 / 3	153 / 13

TABLE II: URL classifier coverage for all SLDs in our dataset, the number of reported categories, and the categories related with ATS services. We identify domains as ATS-related if they are categorized by any service as a variant of the following terms: Ads, Advertising, Analytics, and Marketing.

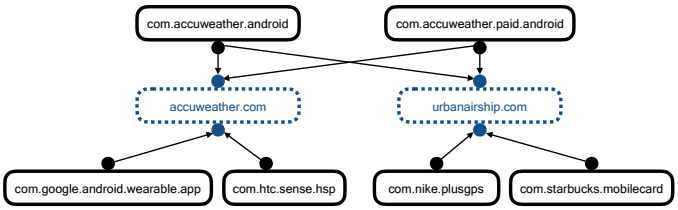

Fig. 2: Communication between six apps (in black) and two online services (in blue).

1) We identify third-party domains by analyzing and comparing the TLS certificates issued by a candidate third-party domain (when available), the flows between apps and domains, and also public information available on Google play. In cases where certificate information is not available (*i.e.*, flows are not TLS-based), we classify the domain as third party if it is contacted by apps from more than one developer.

2a) Using automated search-engine queries and a front-page Web scraper, we identify the third-party domains (services) suspected of providing services associated with ATSes. We auto-curate this list by incorporating knowledge from Web-based classification services.

2b) We label domains which receive unique identifiers from user devices but were not identified in the previous step as ATS-C. Since they possess unique identifiers for users, regardless of their current practices, these domains are all capable of performing tracking.

C. Identifying third-party domains

From the flows in our dataset, we obtain the set of domains with which monitored apps communicate. We use the TLS certificates issued by these domains (when available) to identify their controlling organizations[8]. Additionally, we crawl the Google Play store to identify the developers of each app in our dataset. We perform token matching of the certificate-derived owners and the app developers to identify cases where the certificate owner is not the same as the developer of the app. When this occurs, we consider the domain to be a third-party domain. For example, consider the scenario in Figure 2. Here, the organization derived from the TLS certificate issued by the domain `accuweather.com` only matches the name of the developers of the two Accuweather apps. Therefore our approach considers the domain to be a third-party domain for the other two apps. Similarly, `urbanairship.com` is identified as a third-party domain for all four apps connecting to it.

We note that this condition is more strict than simply comparing the primary domain of the app with domains it communicates with (as has been done in prior work [9]). This is to account for apps that embed content from other domains (e.g., Facebook Like button, YouTube videos). However, it falls short in distinguishing cases where the TLS certificate is owned by a CDN (*e.g.*, CloudFlare) or when the domain does not have any TLS certificates issued. For both of these cases, we consider a domain to be a third-party domain if apps from more than one developer connect to it. This allows us to identify a larger set of third-party domains, but will miss less popular third parties that are only used by a single app developer.

[8]We extract them from the *Organization* or *Organization Unit* fields when the domains are not hosted on popular CDNs which often provide certificates with their own organization names.

D. Developing a classifier to identify ATS domains

We develop a classifier to automatically identify ATS domains from our list of third-party domains. We train the classifier using text from:

1) 2,000 domains from the Alexa Top websites global list [48]. These domains are unlikely to be associated with advertising and tracking.
2) 2,000 randomly sampled domains from Easylist [43]. These domains are suspected to be associated with advertising and tracking on the Web.

We build a Selenium-based scraper using the Firefox browser to obtain data used as input to our ATS classifier. This scraper does the following for each domain 1) visits the front-page of a the domain; and 2) issues "`about <domain>`" queries to the DuckDuckGo search engine [49]. We use the text collected by the scraper for classification.

Data pre-processing. We perform several transformations on the text before classification. First, we remove English stop-words and tokenize the texts into words, bi-grams, and tri-grams and use them as features. We use one hashing vector [50] per domain to count the occurrence of each feature and then normalize the vector using the l_2-norm.

We evaluate multiple popular text classifiers and find that the Linear-kernel SVM classifier has the highest accuracy with an F1-score of 0.95 (precision: 0.95, recall: 0.95) when differentiating between the Easylist (ATS) and Alexa (non-ATS) domains. We use this model to classify our identified third-party domains as ATS and non-ATS.

Post-classification processing. To improve the performance of our classifier, we need to account for noise in the training data (*e.g.*, EasyList contains `bbc.co.uk` and several other popular news sites) and unresponsive sites. Therefore we leverage knowledge gathered from multiple external domain categorization resources. Specifically, we verify that identified ATS domains are not categorized in ATS-unrelated categories by the McAfee [45], OpenDNS [46], and Virus Total [47] categorization services and eliminate those that are. Similarly, we add to our ATS list any domains that are tagged as non-ATS by our classifier, but are categorized in ATS-related categories by all three URL categorization services. This

additional step helps us automatically reduce the frequency of mis-classifications. However, its effectiveness is still dependent on the accuracy of the external URL categorization services.

Validation. To get an understanding of the accuracy of the completely automated ATS-identification process detailed above, we perform manual inspection of 200 domains: 100 randomly-chosen ATS-classified domains and 100 randomly-chosen non-ATS-classified domains. Our inspection revealed that our approach resulted in a 4% false-positive rate and 10% false-negative rate. We found that in all cases false-positives were due to vague categories in the categorization services; *e.g.*, subdomains hosted on `akamaihd.net` were classified as ATS due to either being unresponsive or not returning a human-readable website (*e.g.*, a REST API) when queried, preventing us from using the website front-page scraping method to classify them, and the listing of `akamaihd.net` as a "marketing" (a ATS-related category) domain by the VirusTotal service.

E. Identifying ATS-C domains

Mobile devices host a variety of unique hardware- and user-identifiers (**UIDs**) as those listed in Table I. These UUIDs, except for the Android ID, cannot be changed or reset by the user. Any third party library, even ATS-C libraries whose primary purpose is not providing advertising and tracking services, can piggyback app permissions to access UIDs—or any other permission-protected data—or obtain them via side-channels (without user consent)[9] to track the user activities across different apps on the same device. Therefore, in order to understand the mobile ATS ecosystem completely, we also focus on identifying ATS-C domains. In Section V-E we explore the privacy policies of several of these organizations to understand if they sell or share the gathered UIDs.

We leverage Lumen's ability to detect and report the presence of UIDs in traffic payloads to identify ATS-C domains from those previously identified third-party domains. We label the UID-harvesting domains not previously classified as ATS domains, as ATS-C. Although ATS-C domains identified in this step are capable of performing targeted advertising and user tracking, we do not add them to our ATS list because UIDs may be harvested for non-ATS purposes such as preventing fraud and abuse [52]. However, when UIDs are used for tracking, they are a strong signal for distinguishing Web-specific trackers from those providing Android-specific libraries. This is due to the inability of web trackers to gain access to such system information.

F. Summarized results

Using our approach, we identified 8,099 third-party domains and 2,121 ATSes, from the set of all 40,553 domains. Of the identified ATSes, 233 were previously unreported in any of the popular domain categorization services and Web-based ATS domain blacklists. We found 2,552 of all domains harvesting one or more of the UIDs listed in Table V. Of these, we identified 730 as ATS-C and 306 as ATSes–*i.e.*, 39.9% of the UID harvesting domains were third party domains. Table III shows the number of domains identified in each category. We attribute the high number of non-ATS domains that collect UIDs identified by our system to a combination of it's false-negative rate (10%) in identifying ATSes and the fact that UIDs can be harvested for non-ATS-related first- and third-party-related activities.

[9]During the course of our investigation, we discovered during this study that the undocumented `getprop` command [51] can be used by developers to obtain a list of identifiers including the device serial number, fingerprint, and MAC address. On reporting the use of the command for the purpose of user tracking to the Android development team, they indicated that the command was working as intended and that blocking consent-free access to identifiers via the command would be considered as a feature request in future releases.

Domains	Third-parties	UID-harvesting third-parties	ATS	ATS-C
40,553	8,099	1,019	2,121	730

TABLE III: Number of domains identified in each category.

	ATS overlap 2,121 (100%)	ATS-C overlap 730 (100%)
McAfee	451 (21.0%)	15 (2.0%)
OpenDNS	780 (36.0%)	11 (1.0%)
VirusTotal	1,081 (50.0%)	62 (8.0%)
EasyList	818 (38.0%)	176 (24.0%)
hpHosts	1,652 (77.0%)	258 (35.0%)

TABLE IV: Overlap between our ATS/ATS-C lists and popular categorization services and Web-based ATS blacklists.

Comparison with Web-based ATS lists. In order to understand how our ATS list performs, we make comparisons with the following three popular domain categorization services–McAfee, OpenDNS, and VirusTotal–and two popular Web-based ATS blacklists–EasyList and hpHosts.

We check how many of our identified ATSes were in an ATS-related category in the each of the categorization services, and listed in one of the blacklists. Table IV shows the amount of overlap between our lists and each of the five services. As one might expect, we see most overlap with the Web-based ATS blacklists from EasyList (38.0%) and hpHosts (77.0%). We find that 233 domains in our list do not overlap with any of the services. This is due to the absence of ATSes geared towards mobile users like `urbanairship.com`, `mobiquitynetworks.com`, and `presage.io` in the Web-based datasets.

UID harvesting and ATS-C domains. Table V shows the fraction of domains that harvest each type of UID. We find that third-party domains, representing only 20.0% of all domains, are responsible for a disproportionate fraction (39.9%) of all UID harvesting. Interestingly, we also find that only 14.4% of all ATSes harvest UIDs from the device, suggesting the use of other techniques such as HTTP headers, cookies, and tracking pixels for tracking. The most common value harvested by ATSes is the semi-persistent Android ID. Interestingly the AndroidID is also collected by ATS-C domains along with at least one persistent UID in 34% of all cases. In addition to making it possible for ATS-C services to persistently track a user, this behaviour contradicts Android's developer policy center guidelines which states that the Android ID should not be associated with any other personally-identifiable informa-

UID	FQDNs N = 40,553	TPs N = 8,099	ATS N = 2,121	ATS-c N = 730
IMSI	201 (0.5%)	89 (1.1%)	13 (0.6%)	76 (10.4%)
IMEI	188 (0.5%)	101 (1.2%)	33 (1.6%)	68 (9.3%)
SIM#	340 (0.8%)	181 (2.2%)	55 (2.6%)	126 (17.3%)
AndSerial	184 (0.5%)	134 (1.7%)	55 (2.6%)	79 (10.8%)
AndId	1,611 (4.0%)	597 (7.4%)	173 (8.2%)	424 (58.1%)
Phone#	102 (0.3%)	50 (0.6%)	8 (0.4%)	42 (5.8%)
Fingerprint	186 (0.5%)	129 (1.6%)	25 (1.2%)	104 (14.2%)
MAC	133 (0.3%)	73 (0.9%)	17 (0.8%)	56 (7.7%)
AnyUID	2,552 (6.3%)	1,019 (12.6%)	306 (14.4%)	713 (97.7%)

TABLE V: List of UIDs monitored by Lumen and the percentage of harvesters in each category.

tion [53]. Another major concern is that the IMEI, a persistent value which uniquely identifies a mobile device, is the fourth most commonly harvested UID and is disproportionately gathered by ATS and ATS-C domains.

V. Illuminating the Mobile ATS Ecosystem

We now investigate the characteristics of the ATS and ATS-C domains identified by our classifier. Specifically, we focus on uncovering the organizations that own and operate these domains, how they cooperate with each other, how often they track users across different devices and platforms (*i.e.*, cross-device tracking), and their pervasiveness in different app categories.

A. ATS popularity and parent organizations

Many ATS domains may belong to the same parent organization. For example, in Figure 3, which shows the number of apps that use the 20 most popular ATS and ATS-C services, we see that 16 of the 20 most pervasive ATS and ATS-C domains are owned by Google's parent company Alphabet. Since subsidiaries of an organization may share data with one another (as we demonstrate in Section V-E), it is important to explore these relationships. To uncover these parent-child organizational relationships in depth and for all domains in our ATS and ATS-C lists, we use the Crunchbase [54] and D&B Hoovers [55] databases. These databases provide access to complete organization structures and information about company acquisitions and mergers. In the event that the owner of the domain cannot be identified accurately via these databases, we resort to obtaining organizational information via any TLS certificate associated with it (when available).

Overall, we identify 292 parent organizations that own nearly 2,000 ATS and ATS-C domains. We find that some third-party services register subdomains within large cloud services or use pseudo-random names through proxies to preserve their anonymity. This behavior has consequences for our parent organization identification approach as it may fail to properly associate such domains with the actual advertising and tracking service. For example, although the domain `supersonicads-a.akamaihd.net` is hosted on Akamai, it is actually operated by SupersonicAds. The problem is compounded with non-TLS traffic since we cannot leverage TLS certificate information to extract parent organization information. For such cases, we label the parent organization as *unknown*.

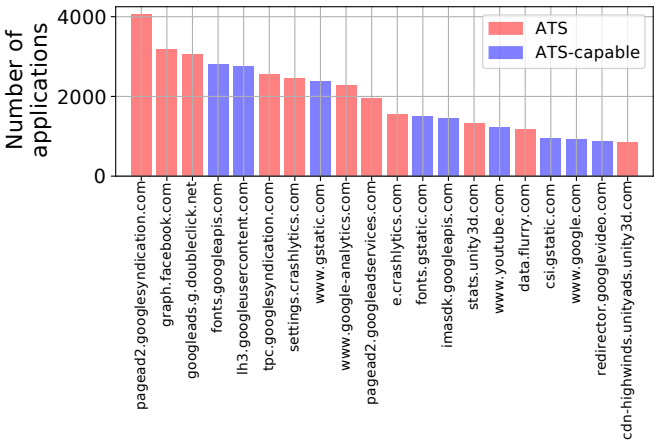

Fig. 3: Number of apps using the 20 most popular ATS and ATS-C services.

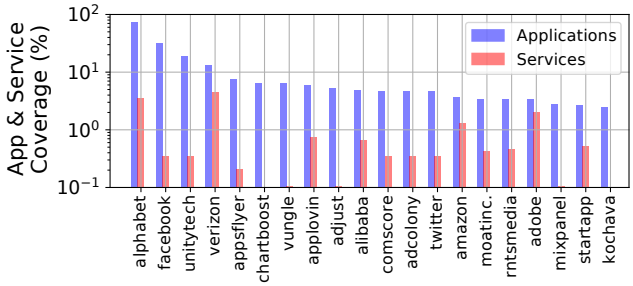

Fig. 4: **(log-scale)** Organizations that provide ATS-related services to the largest fraction of apps and the fraction of services owned by them.

Figure 4 shows the 20 organizations with the largest app penetration and the fraction of services owned by them. We find that Alphabet has penetration in over 73% of all our measured apps with ownership of only 3.6% of all ATS and ATS-C services. Facebook–known by average users for providing social networking services–has ATS presence in over 31% of all measured apps while owning only 0.35% of all ATS and ATS-C services through the Facebook Graph API. Interestingly, Verizon Wireless provides ATS services to 13.1% of our measured apps with *all* the services coming through acquisitions of AOL and other ATS vendors. We see a similar trend with Adobe which has presence in 3.3% of all measured apps. Our analysis shows that, of the Top 10 most dominant organizations in the ATS ecosystem, only Chartboost, Vungle, and Adjust provide ATS-related services as their primary business. In general, we find that ATS-specialists appear to have a competitive market share while owning only a handful of services, as illustrated by the larger differences in the app and service coverage percentages in Figure 4.

B. Analyzing ATS domain co-occurrences

We now focus on identifying which ATS domains are likely to operate simultaneously in similar apps. This may occur

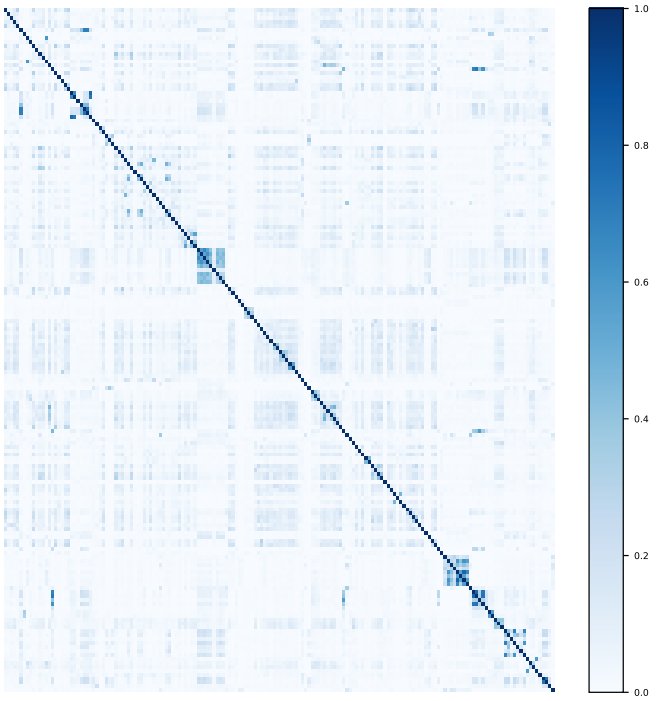

Fig. 5: Jaccard similarity between the app penetration of pairs of ATS domains present in at least 50 apps. Domains are sorted by parent organization. Clusters of co-occurring domains show domains owned by the same organization are more likely to be present in the same app.

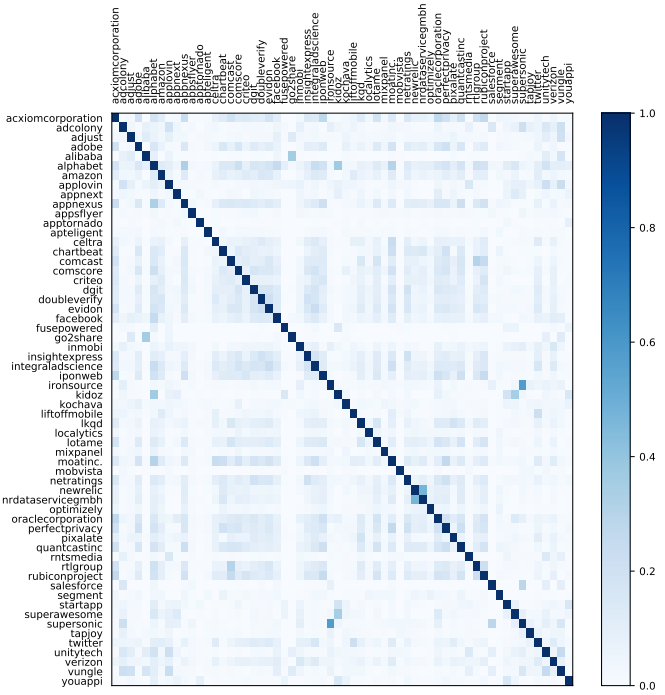

Fig. 6: Jaccard similarity between the app penetration of pairs of the largest parent organizations.

due to developers bundling several ATSes in their app or using SDKs which bundle services from multiple domains. Frequently co-occurring domains may also be used as a signal that the domains are owned and operated by the same parent organization. We leverage the Jaccard similarity index for the purpose of identifying strongly co-occurring domains. Specifically, $JS(dom_a, dom_b) = \frac{apps_a \cap apps_b}{apps_a \cup apps_b}$. Here, $apps_x$ denotes the applications that are observed to interact with the domain dom_x.

Figure 5 shows the Jaccard similarities for all pairs of domains that were used by at least 50 apps. The results are sorted by their parent organization (when this information is obtainable). We find that our hypothesis that domains belonging to the same organization are more likely to co-occur is valid – *i.e.*, we see an average similarity score of 0.15 for such domains vs. 0.03 for domains belonging to different organizations.

We use this insight to associate parent organizations to domains having pseudo-random identifiers and those hosted on popular CDNs. As an example, we find that the domain `amazon-d3v1lb83psg9di.cloudfront.net` has the highest co-occurrence with other AdColony owned domains as did `supersonicads-a.akamaihd.net` with other Supersonicads domains (both relationships had $JS > 0.7$). We were also able to uncover previously unknown (to our database of organizations) relationships between ironSource and Supersonicads, and the anonymously registered domain `iasds01.com` to the IntegralAdScience ATS vendor. All of the above findings were confirmed via manual inspection of publicly available information on the Web.

We also leverage the similarity index to identify clusters of organizations that occur in similar sets of apps, giving them access to similar types of audiences. For example, we find that domains from key players such as Alphabet (specifically, Doubleclick and Crashlytics), MoatInc., and Facebook have high similarity scores. We identify a second cluster of organizations that provide ATS-related services specifically to the mobile gaming market—AppLovin, Vungle, and Unity3D. Other strong relationships occur between OnlineMetrix and TapJoy, and Ilyon Analytics and Supersonicads. These organizational co-occurrences are fully illustrated in Figure 6.

C. Application characteristics and ATSes

Mobile apps may use multiple advertising and tracking services simultaneously. For instance, developers may combine different ad networks to maximize their revenues [12]. Figure 7 shows the distribution of the number of domains reached per app according to 4 categories: all domains, ATS, ATS-C, and domains falling in any other category. The average mobile app connects to 11 different domains, out of which 4 and 2 are ATS and ATS-C domains, respectively (see Figure 8 for median values). The analysis reveals that ATS and ATS-C domains are almost as prevalent as other domain categories on mobile apps: 82 % and 29 % of them connect to at least one and 5 ATS domains, respectively. The percentage of apps connecting at least to 1 and 5 ATS-C domains is 75 % and 29 %, respectively. Interestingly, only 40 % of the apps voluntarily report the presence of ads with the *Contains Ads* label on their

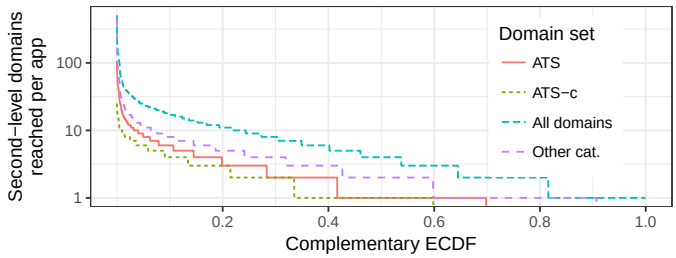

Fig. 7: (log-scale) Distribution of the number of ATS, ATS-C and other domain categories per app.

Google Play profile [10].

Impact of app pricing model. We analyze the presence of ATS and ATS-C domains per app monetization model (*i.e.*, paid, free, and free apps with *in-app purchases*). Free apps with in-app purchases connect on average to 3 ATS services and 2 ATS-C. This ATS and ATS-C penetration is higher than on completely free apps: 2 and 1, respectively. Although paid apps are not free from tracking, they appear to have the lowest number of trackers on average: 1 ATS and 1 ATS-C domain. 23 % of paid apps have no tracking activity at all as opposed to 12 % and 7 % in the case of free apps and free apps with in-app purchases respectively. This suggests that apps with in-app purchases may have more aggressive monetization strategies.

Impact of app category. Figure 8 shows the number of ATS and ATS-C services per app category. Surprisingly, games and educational apps are the two categories with the highest number of ATS and ATS-C domains. Our analysis allowed us to identify ATS and ATS-C services specialized in providing support to game developers such as `everyplay.com`, `playfab.com`, `gameanalytics.com`, and `mindjolt.com`. Of these, the ATSes provided by Unity3D and Vungle are the most prevalent in mobile gaming apps. As we can see by the distribution outliers, users from news and entertainment apps may also be exposed to a wide range of ATS and ATS-C domains. We hypothesize that this is due to the presence of traditional Web trackers embedded in the content rendered by such apps (mainly comScore and Google trackers).

D. Cross-device tracking.

We now investigate which of these organizations and ATSes may have the ability to perform cross-device tracking. For that, we analyze whether they have any presence in the Web ecosystem. The ability to perform cross-device tracking would allow them to link mobile app and Web usage behavior and possibly reveal a very privacy-invasive insight into an individual's virtual and real-world habits. To that end, for each ATS and ATS-C domain, we measure their penetration in the Alexa Top 1,000 websites as third-party Web trackers. We crawl the desktop versions of the Alexa Top 1,000 websites using the Firefox browser with Selenium.

In our analysis we find that 39% of all our identified ATSes are present as third-parties in at least one of the Alexa

[10]Google Play does not require developers to report the presence of third-party ATS services embedded in their apps and the organizations behind.

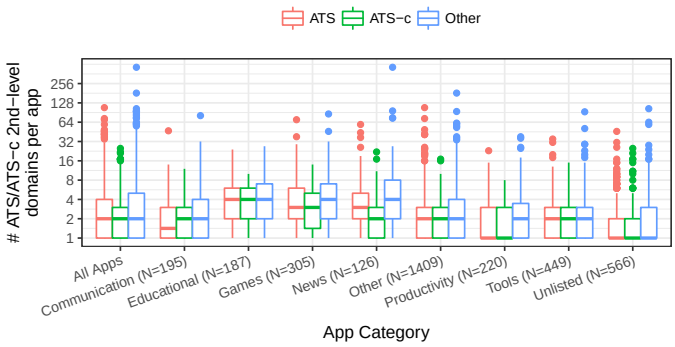

Fig. 8: **(log2-scale)** Boxplot with the distribution of ATS, ATS-C and other domain categories reached by apps from selected Google Play categories.

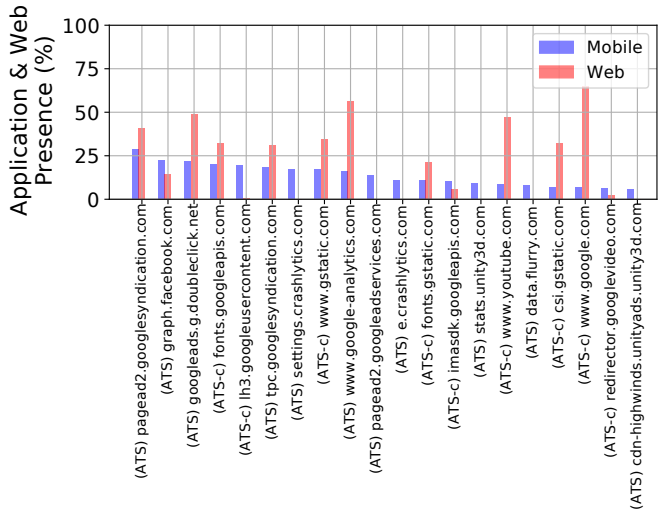

Fig. 9: Presence of the most popular ATS and ATS-C services on measured apps and the Alexa Top 1,000 websites.

Top 1,000 websites. Additionally, when only considering the top 20 mobile ATS and ATS-C services, only domains from Crashlytics, Flurry, and Unity3D were absent from the Web. Figure 9 shows the pervasiveness of the 20 most popular mobile services on the Web. From our top 20 ATS parent organizations, only Chartboost, Vungle, and Adjust were absent from the Web. These findings show that cross-device tracking is already widespread: even organizations that do not specialize in ATS-related services have presence in both platforms.

E. Privacy and data sharing policies of ATSes

To understand the privacy impact of user tracking, we need to study data sharing practices of ATS organizations and find out which entities will ultimately have access to user data after it has been collected. Unfortunately, there is very little information available publicly about ATS companies sharing or selling user data. However, while with our current framework, we are unable to detect and analyze flows of information from parent organizations (gathered in their roles as third-parties) to their subsidiaries and partners, companies providing ATSes publicly outline their policies regarding collection and sharing

of tracking data, their partnerships with other companies and services, and subsidiaries in their public websites and privacy policy documents; which we can use as a proxy to understand the scope of such data sharing practices. Studying this public information helps us uncover data sharing relationships between ATSes, entities that can ultimately access tracking data beyond the original collecting company or service, and what the end users can do to opt out of collection of tracking data or its use for targeted ads. Due to the difficulty in scaling the manual effort involved in extracting relevant points from privacy policies of each parent organization, we focus our analysis of privacy policies on the 10 largest ATS parent organizations, whose ATSes are present in over 82% of all apps in our dataset and whose policies have the largest impact on user privacy.

Table VI shows key points extracted from privacy policies and public websites from the top 10 ATS parent organizations. We find that each of these organizations operate at least one advertising and one tracking service. Further, each of their privacy policies specifically state that they allow sharing of tracking data with their subsidiaries (even AppsFlyer and Adjust which have no publicly known subsidiaries). More worrying is the fact that, with the exception of Alphabet and Facebook, all organizations also reserve the right to share their ATS-related data with third-party partners. Therefore, developers who use services provided by these organizations provide a gateway for more third-party organizations to track their users.

While some of these organizations provide end users with a way to opt out of using tracking data to display targeted ads, none allow the users to entirely opt out of tracking or having their data shared with other organizations. Even when opting out of targeted marketing, users face different procedures from different organizations. The organizations which are part of the Network Advertising Initiative (NAI)[11] and the Digital Advertising Alliance[12] allow end-users to use web forms provided by these organizations to opt out of interest-based and targeted ads. However, others such as AppsFlyer, Vungle, and Alibaba have more convoluted approaches – *e.g.*, email and webforms specific to their organizations.

VI. User Tracking: A Regulatory Challenge

Our analysis so far has shown that user tracking is already pervasive in the mobile platform and even across platforms. Further, the widespread sharing of tracking-related data with subsidiaries and third-parties, combined with (often) complicated and non-standard procedures for opting out of interest-based marketing has significantly reduced user control and authority over their own data. In this section, we explore the impact of upcoming regulations on UID harvesting and recent regulations on tracking vulnerable audiences.

A. ATSes: A global regulatory problem

In order to understand how frequently ATS-related data flows across borders and jurisdictions, we identify the locations of the sources (Lumen users) and sinks (ATS-related IP addresses) of the flows in our dataset. We use the MCC reported by Lumen to geolocate users and Maxmind database [56] to geo-locate ATS servers. Our analysis reveals that the United States hosts over 40% of all ATS servers and is at the terminating end of over 50% of all cross-border ATS traffic. We also find that China and South Korea are hosts to over 9% of all ATS servers (a majority of which are owned by Alibaba and IGAworks). In Figure 10, we display the fraction of flows observed to be between Lumen users and ATS servers located in each country. We find that the ATS servers in the United States have disproportionately higher access to ATS-related data – *i.e.*, although only 40% of the global ATS servers are housed in the country, they are at the terminating end of 73% of all ATS-related flows.

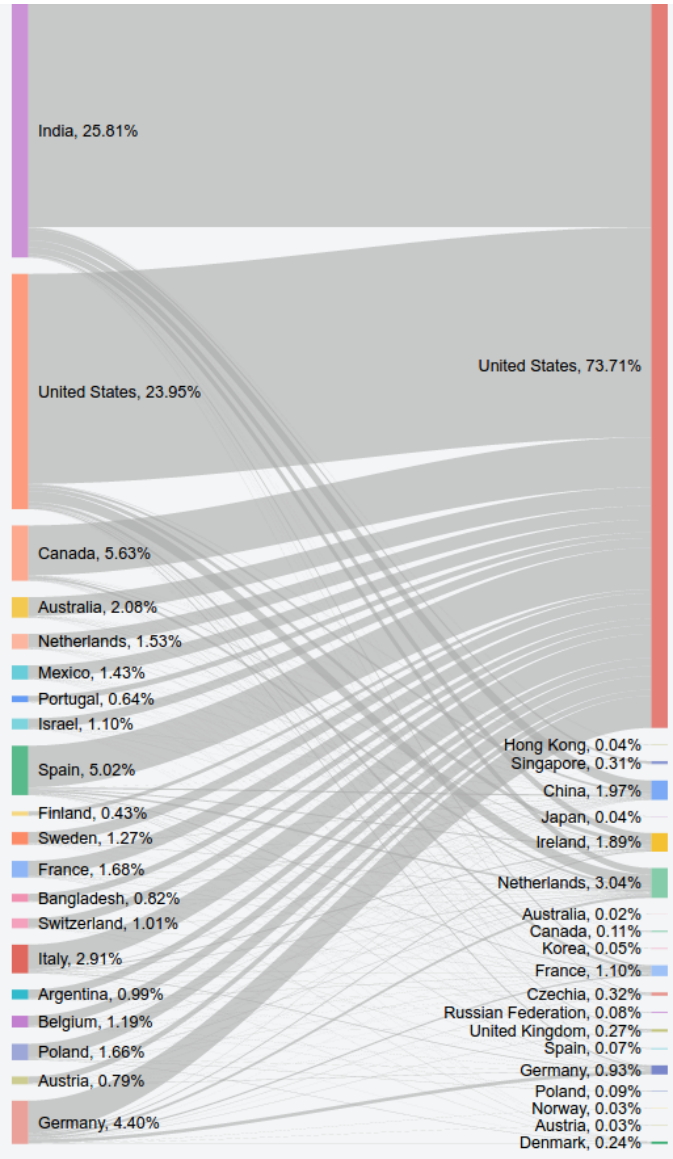

Fig. 10: Interactions observed between the 20 most common locations of Lumen users and ATS server locations. Percentages indicate the fraction of flows originating (or, terminating) at the corresponding country.

We also find that even users from countries with strong consumer and privacy protection laws (*e.g.*, Switzerland, Ger-

[11] https://www.networkadvertising.org/
[12] http://digitaladvertisingalliance.org/

Parent	Has subsidiaries?/Data shared with subsidiaries/3rd parties?	Opt-out procedure	COPPA/Childrens policy
Alphabet	✓/ ✓/ ×	✓(via account settings)	ATSes not directed at children, has complaint email address.
Facebook	✓/ ✓/ ×	✓(via account settings)	No policy.
Twitter	✓/ ✓/ ✓	✓(via account settings / DAA webform)	No policy.
Verizon	✓/ ✓/ ✓	✓(via account settings)	Full policy based on parental consent.
AppsFlyer	× / ✓/ ✓	✓(via email)	ATSes note directed at children, has complaint email address.
Chartboost	✓/ ✓/ ✓	✓(via NAI/DAA webforms)	No policy, up to developers.
Vungle	✓/ ✓/ ✓	✓(via Google ID reset)	No policy, up to developers.
AppLovin	✓/ ✓/ ✓	✓(via TRUSTe/EU YOC)	Does not collect data from children apps, has complaint system but no email address.
Adjust	× / ✓/ ✓	✓(via NAI webform)	No policy, has complaint email address.
Alibaba	✓/ ✓/ ✓	✓(via webform)	No policy.

TABLE VI: Privacy policy and public website information from top 10 most dominant ATS providers.

many, and Spain) have sizable fractions of ATS-related traffic flowing into nations with weaker regulatory frameworks. Such trans-national flow of data makes it unclear which privacy and consumer protection laws are applicable to ATS-related data.

To address this problem, the European Union is making progress to define regulations targeting how personal information and metadata may be collected and used for marketing: the General Data Protection Regulation (GDPR) [57] and the ePrivacy directives [58]. The GDPR rule aims to control how organizations collect and store any personal data that can, directly or indirectly, identify European citizens. This definition includes information such as personal names, religion, addresses and biometric information as well as digital information and identifiers like email addresses or IMEI codes.

Additionally, whenever user's sensitive data is sent over telecommunication services like the Internet, the European Directive 2002/58/CE (a.k.a. the ePrivacy Directive) also applies. This directive obligates entities that collect user data to inform the users about the third-parties who will receive this data, and requires them to ask for their consent whenever they intend to use the data for purposes other than the original purpose for which it was collected. The ePrivacy Regulation, currently being debated at the European Union, will replace this directive. However, the final text of the ePrivacy regulation is unknown as of this writing. In fact, it is not expected to be adopted until May 2018 [13]. As a result, it remains unclear how both rules will work together to protect user privacy. Below, we summarize the expected high-level impact of these regulations on ATSes and mobile apps:

- **Applicability.** The regulations are aim to protect the privacy of any individuals residing in the EU and are directed at organizations (including those based outside the EU) who gather information, digital or not, that may be used to directly or indirectly identify individuals. The current terminology suggests that all previously identified ATS and ATS-Cs will be subject to this regulation.
- **Consent.** Any data that may be used to directly or indirectly identify an individual must explicitly request user consent – *i.e.*, data harvesting requires users to opt-in. Consent has to be explicit and must specify the data being gathered and the purpose of such gathering. For individuals under the age of 13, verifiable consent must be obtained by a guardian. It is unclear if the current approach used by Android (requesting install-time permissions from users) is sufficient for the purpose of gathering consent. Additionally, it is unclear if the responsibility of gathering consent falls to the third-party ATS which receives the data or the mobile app which provides access to it.
- **Retention.** The directive specifies that except for cases when the gathered data is essential to the functioning of the service (*e.g.*, cookies for shopping carts), personal data must be stored in a way that anonymizes the subject. An exception is when the data is gathered with the user's informed consent for providing value-added services (*e.g.*, targeted ads). This consent, however, may be withdrawn by the user at any time. When consent is withdrawn, all data pertaining to that individual is subject to erasure. Further, subjects are required to have access to their harvested personal data. While these retention policies significantly increase consumer protection, they do not regulate specifically ATS consent withdrawal and data access procedures. This may lead to non-uniform and sometimes inefficient procedures – *e.g.*, requiring subjects to make withdrawal requests via email. Table VI shows the different ways in which ATS organizations currently implement opt-out options: some companies rely on the OS-provided settings to facilitate opting-out of interest-based advertisements, while others use arguably less efficient and nonintuitive methods such as webforms and complaint email systems.
- **Penalties.** Organizations that do not comply with the proposed regulations may be issued warnings (for first offenses) and face fines of up to 1 Million Euros (or 2% of global revenue) for repeat violations.

Since these regulations also apply to organizations operating outside the European Union, we seek to understand how common it is for UIDs of EU residents to be harvested by ATSes outside the European jurisdiction. Figure 11 shows the source and destination countries of flows containing UIDs. We find that ATSes hosted in the United States and China are likely to be the most impacted by the upcoming regulations. Our methodology is limited by its inability to identify how and when user consent is obtained and how harvested data is stored by ATSes, and is therefore unable to identify ATSes that are (or, will be) in violation of these regulations. However, we note that our analysis of privacy policies of the most dominant

[13] http://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX

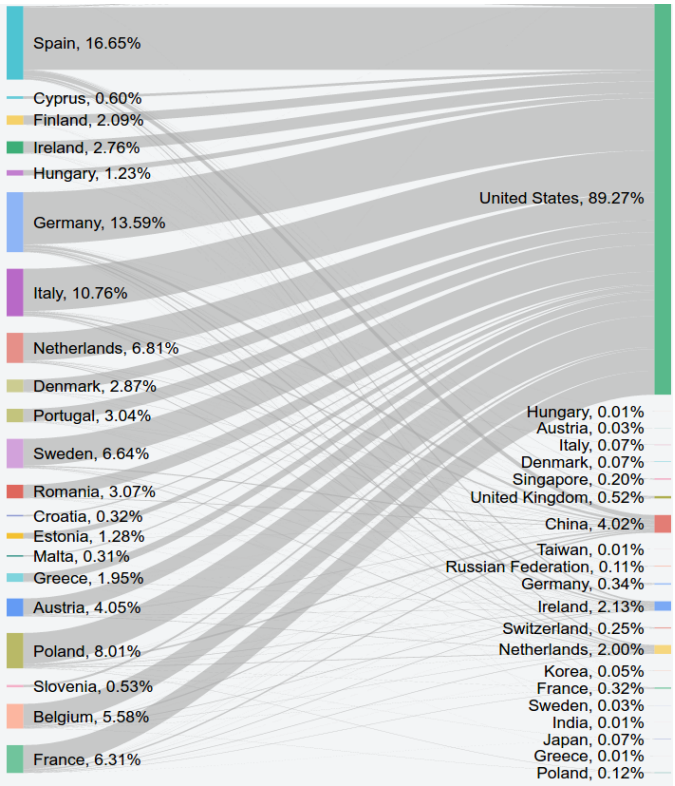

Fig. 11: Flow of UIDs from nations in the European Union. Percentages indicate the fraction of flows originating (or, terminating) at the corresponding country.

ATS organizations suggests that they all at least claim to be in compliance with current regulations in the EU.

One major shortcoming of the regulations being proposed is the absence of strong limitations on how harvested data may be shared with subsidiaries and third party organizations. Since the regulations are still being drafted, we hope that our work will motivate the inclusion of such policies.

B. Trackers targeted at vulnerable audiences

As shown in Section V-C and Figure 8, we uncover a large number of apps containing ATSes in the gaming and educational categories. In the United States there are a number of restrictions on how tracking may be performed in products directed at children. These are specifically regulated in the Children's Online Privacy Protection Act of 1998 (COPPA) [28]. COPPA mainly seeks to enforce that verifiable consent of a guardian is obtained before harvesting data of a child (under the age of 13) and that data is not retained for a longer duration than is necessary for providing the requested service.

In our analysis, we find that 88 % of the games and education apps in our dataset may be required to be COPPA-compliant–*i.e.*, they are labeled as suitable for children under the age of 12 according to their ESRB rating [59]. However, 24 % of them use at least one ATS or ATS-C service and leak unique identifiers to them. A preliminary investigation of the privacy policies of the ATSes present in these apps shows that a large number do not outline any policy addressing their use in apps targeted at children and special data handling for UIDs obtained from children. This is potentially in violation of the COPPA rule. Surprisingly, many of largest ATS parent organizations either do not have a privacy policy or Terms and Conditions clause addressing children, have generic clauses stating that they do not knowingly collect tracking data from minors, or imply that it is the responsibility of app developers to make sure their services are not used by minors without parental consent (Table VI).

We note that our current methodology does not allow us to investigate possible COPPA violations in depth. To do so, it is necessary to check whether each app requests parental consent during runtime and the specific content of their (app and ATS) privacy policy – a promising avenue for future work. A preliminary investigation of ATSes geared at children's apps shows that many of them (*e.g.*, Kidoz) have specific provisions in their policies and claim to be in compliance with COPPA. However, at the time of writing, Disney was sued for COPPA violations in 42 of their apps [30].

VII. Discussion

While new regulations for online marketing and data harvesting (such as the previously discussed GDPR and ePrivacy directives and COPPA) provide avenues for prosecuting mishandling of sensitive data, they fall short in several ways:

- Due to the opacity of the tracking ecosystem, it is difficult to uncover and track how organizations collect personal data from end users, and how they store and share it with each other.
- The regulations leave room for interpretation in several cases. Specifically, they do not describe exactly how consent must be obtained from users – *e.g.*, it is unclear if gaining permissions to access UIDs at install time is sufficient – and how consent withdrawal should be facilitated – *e.g.*, is it sufficient for a user to uninstall an app, or even if explicit notification is required, should easy to access unified web forms be made available (as is the case with the DAA).
- While some ATS-related organizations are more responsible than others and have comparatively more reasonable data sharing policies, current regulations do little to limit the sharing and selling of data by these organizations, leaving users with almost no control of who has access to their data.

While our focus is not to find instances of data being shared or sold by organizations, examining the privacy policies of the key players in the mobile ATS ecosystem reveals data sharing policies to be very prevalent, with 8 out of 10 ATSes present in over 82% of mobile apps reserving the right to share tracking data with third-parties, a major privacy concern that is not known by end-users of mobile apps.

As in the Web ecosystem, a great number of developers rely on ATS services to monetize their apps. Privacy control tools such as anti-tracking services and ad-blockers have been recently the subject of a great deal of discussion due to their disruptive nature and interference with the economic sustainability of mobile apps and online services [60]. While app developers sometimes acknowledge the fact that a great number of their users do not want to be tracked, and offer them a way to disable targeted advertisements by purchasing paid

versions of the same apps, many paid apps still include third-party services such as analytics services that collect unique identifiers, and many developers do not provide any form of control or transparency over how user data is collected, stored and shared with third party services. App developers and organizations that regulate app stores have a responsibility to provide more transparency and user control to end-users, least the mobile app ecosystem enters an arms race similar to the one between publishers and consumers on the Web.

VIII. Limitations

We note that each of our contributions has its own limitations. First, even though our ATS classification approach is able to identify 233 previously unknown ATSes that operate in the mobile ecosystem, it is not completely accurate—some ATS and ATS-C domains may not have been identified— and relies on noisy external data from Web-specific ATS blacklists and categorization services for training and post-processing (Section IV). Unfortunately, the scale of the ecosystem makes manual sanitization infeasible. Second, although our analysis of the major organizations engaging in ATS-related activities unveiled the pervasiveness of tracking (mobile and cross-device) and the commercial relationships between popular ATSes, it has the shortcoming of being unable to perfectly identify the owners of ATSes hosted on popular CDNs and with pseudorandom identifiers in their domain names.

Measuring the real impact of tracking activities is difficult especially given that privacy depends on individual factors and context. It is beyond the scope of this work to measure the real impact for the users. What can be done to inform the user and report data flows, however, is to develop privacy-enhancing tools to show users which entities collect information about them and whether the entities collecting this information reserve the right to share that information with other parties (given that they have a privacy policy). The user can then decide based on this information and other individual factors whether they want to continue using apps that include certain ATSes or not. Finally, in our analysis of the impact of the GDPR, ePrivacy, and COPPA regulations, we were not specifically able to identify regulation violations due to our inability to identify when and how user consent was gathered and how user data is handled by ATSes (Section VI). Nevertheless, we were able to provide previously unknown insights into the operation and pervasiveness of tracking in the mobile ecosystem, opening new venues for future research.

IX. Future Work

Our analysis reveals the need for further investigation in several directions. One promising avenue includes understanding business relationships between organizations (beyond the parent-child relationships identified in this work) while performing deeper and larger-scale investigations of the privacy policies of ATS providers and app developers. Such a study might more clearly uncover the scale of data sharing between seemingly unrelated organizations. Next, our inability to specifically identify violations of the GDPR, ePrivacy, and COPPA regulations occurs due to the absence of data regarding how apps gather user consent. In future work we plan to utilize a lab test-bed to analyze how consent is gathered by apps that are known to harvest UIDs from the audiences protected by these regulations and explore the use of tools such as Privee [61] to automatically analyze privacy policies.

Finally, we also plan to use the results of our study to extend the functionality of the Lumen app to provide users the ability to block flow of UIDs from their devices to specific domains. As such, our vision is not blocking or modifying any ATS flow by default, instead giving users the knowledge and the power to make informed decisions by themselves to prevent abusive practices. Our goal is to empower mobile users, and not to weaken developers' position.

X. Related Work

Compared to similar studies that focus on detecting and analyzing ATSes in the Web [5]–[8], studying mobile ATSes is particularly challenging due to a lack of data collection methods to enable monitoring of mobile apps at a large scale. The research community and regulatory agents have leveraged various techniques to detect privacy violations inflicted by mobile apps and to identify third-party advertising and tracking services:

Traffic analysis. Previous research efforts have used mobile ISP traffic logs to characterize mobile advertising traffic and their aggregated effect on a mobile network [12], and characterizing the economic aspects of the mobile advertising industry and online aggregation services [9]. Likewise, other researchers instrumented VPN servers to redirect and identify privacy leaks on user's traffic [3]. These approaches have used information available in the payload (*e.g.*, the `User-Agent` header value in HTTP flows) of observed flows to infer the originating apps. Unfortunately, traffic logs captured at the network do not provide sufficient context to accurately identify the app originating a given flow, making difficult application attribution. To overcome this limitation, various research teams developed solutions such as Lumen Privacy Monitor– the tool implemented for this study – Privacy Guard [35] and AntMonitor [34] to analyze mobile traffic on the app itself using the Android VPN permission.

Dynamic Analysis. This second class of prior work aims to understand app behavior using dynamic analysis of app binaries in controlled environments. Dynamic analysis calls for running an app in a highly controlled environment such as a virtual machine [62] or instrumented OS [4], [63]. The app is then closely monitored as it conducts its set of tasks, with the results indicating precisely how the app and system behave during the test (*e.g.*, whether the app exfiltrated location information). Dynamic-analysis approaches can automate user interaction using UI exercisers, [64]–[66]. However, these techniques are well known for not being able to cover all contexts and scenarios, such as where user-specific inputs are required, as indicated by a previous study [17]. Moreover, the artificial workload –user interaction is typically generated by UI executioners– and difficulty of deploying custom firmware on users' phones means the results do not directly speak to normal users' activity.

Static Analysis. Static analysis involves analysis of the app code through analyzing executables and producing control flow graphs [19], [67], using symbolic execution [18], by auditing

third-party library use [2], or through inspection of the permissions apps demand from the user [68]. While static analysis typically provides good scale with analysis of over 10K apps in many studies, this strategy does not reflect the behavior of apps in the wild with real user stimuli. In particular, static analysis may understate or overstate the importance of certain code paths since it lacks any notion of how users actually interact with their apps in practice. Further, these approaches face challenges when analysing apps whose code has been obfuscated. To deal with these challenges, Continella *et al.* [69] propose a black-box analysis tool to detect privacy leaks in mobile apps even when they use obfuscation. Moreover, apps can potentially download code during runtime which can not be examined via static analysis alone. Backes *et al.* recently studied the presence of third-party libraries (independently of their purpose) in top Android apps [70].

Cross-device tracking. Zimmeck *et al.* [22], in a study aimed at quantifying the occurrence of cross-device tracking, identified 81 mobile ATSes operating in 845 mobile apps by analyzing the types of SDKs utilized by the apps (this was in addition to 3,243 ATSes operating on mobile versions of popular websites). These previous studies have helped to improve our understanding of mobile tracking and its implications on user's privacy. However, their focus was limited to understanding specific apps and trackers, rather than the overall dynamics of the mobile ATS ecosystem.

XI. Conclusions

At a high-level, this study provides a new traffic-oriented perspective to understand the domains and organizations related with mobile advertising and tracking (ATS) activities and their behavior in the wild. We implemented an automated mechanism to identify ATS domains. Our technique allowed us to identify many ATS domains which operate specifically in the mobile ecosystem and were previously unreported by well-known Web-based ATS blacklists like Easylist and other popular URL classifiers. We measured the pervasiveness of the identified ATSes in the mobile and Web platforms. Our investigation also shed light on the relationships between different ATS vendors by identifying the parent organizations of the services. Our privacy policy analysis of the largest organizations revealed the prevalence of intra- and inter-organization sharing of user data. Finally, we considered how ATS-related data (and specifically UID data) flows across borders and the impact of privacy regulations in the European Union and United States on ATSes. We hope that our findings will spark and inform more public discourse and result in stronger regulatory frameworks to protect user privacy.

Acknowledgments

As always, we are deeply grateful to our Lumen users for using our tool and enabling this study. We thank the anonymous reviewers and Moises Barrio for their helpful feedback. This project is funded by the NSF grants CNS-1564329, the European Union under the H2020 TYPES (653449) project and the Data Transparency Lab Grants (2016). Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not reflect the views of the funding bodies.

References

[1] S. Englehardt and A. Narayanan, "Online tracking: A 1-million-site measurement and analysis," in *Proc. of ACM CCS*. ACM, 2016.

[2] T. Chen, I. Ullah, M. A. Kaafar, and R. Boreli, "Information leakage through mobile analytics services," in *Proc. ACM HotMobile*, 2014.

[3] J. Ren, A. Rao, M. Lindorfer, A. Legout, and D. Choffnes, "Recon: Revealing and controlling pii leaks in mobile network traffic," in *Proc. of ACM MobiSys*, 2016.

[4] W. Enck, P. Gilbert, B. Chun, L. Cox, J. Jung, P. McDaniel, and A. Sheth, "TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones." in *Proc. USENIX OSDI*, 2010.

[5] B. Krishnamurthy and C. Wills, "Privacy diffusion on the web: A longitudinal perspective," in *Proc. of the World Wide Web Conference*, 2009.

[6] D. Malandrino, A. Petta, V. Scarano, L. Serra, R. Spinelli, and B. Krishnamurthy, "Privacy awareness about information leakage: who knows what about me?" in *Proc. of WPES*, 2013.

[7] O. Starov and N. Nikiforakis, "Extended tracking powers: Measuring the privacy diffusion enabled by browser extensions," in *Proc. of the World Wide Web Conference*, 2017.

[8] O. Starov, P. Gill, and N. Nikiforakis, "Are You Sure You Want to Contact Us? Quanitfying the Leakage of PII via Website Contact Forms," in *Proc. of PETS*, 2016.

[9] P. Gill, V. Erramilli, A. Chaintreau, B. Krishnamurthy, K. Papagiannaki, and P. Rodriguez, "Follow the money: Understanding economics of online aggregation and advertising," in *Proc. of ACM IMC*, 2013.

[10] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin, "A first look at traffic on smartphones," in *Proc. of ACM IMC*, 2010.

[11] N. Vallina-Rodriguez, A. Aucinas, M. Almeida, Y. Grunenberger, K. Papagiannaki, and J. Crowcroft, "RILAnalyzer: a comprehensive 3G monitor on your phone," in *Proc. of ACM IMC*, 2013.

[12] N. Vallina-Rodriguez, J. Shah, A. Finamore, Y. Grunenberger, K. Papagiannaki, H. Haddadi, and J. Crowcroft, "Breaking for commercials: characterizing mobile advertising," in *Proc. of ACM IMC*, 2012.

[13] Google, "UI/Application Exerciser Monkey," https://developer.android.com/tools/help/monkey.html.

[14] A. Machiry, R. Tahiliani, and M. Naik, "Dynodroid: An Input Generation System for Android Apps," in *Proc. of ESEC/FSE*, 2013.

[15] C.-J. M. Liang, N. D. Lane, N. Brouwers, L. Zhang, B. F. Karlsson, H. Liu, Y. Liu, J. Tang, X. Shan, R. Chandra, and F. Zhao, "Caiipa: Automated large-scale mobile app testing through contextual fuzzing," in *Proc. of ACM MobiCom*, New York, NY, USA.

[16] P. Carter, C. Mulliner, M. Lindorfer, W. Robertson, and E. Kirda, "CuriousDroid: Automated User Interface Interaction for Android Application Analysis Sandboxes," in *Proc. of FC*, 2016.

[17] I. Reyes, P. Wijesekera, A. Razaghpanah, J. Reardon, N. Vallina-Rodriguez, S. Egelman, and C. Kreibich, "is our childrens apps learning? automatically detecting coppa violations," in *Workshop on Technology and Consumer Protection (ConPro 17)*. IEEE, 2017.

[18] Z. Yang, M. Yang, Y. Zhang, G. Gu, P. Ning, and X. Wang, "AppIntent: analyzing sensitive data transmission in android for privacy leakage detection," in *Proc. of ACM CCS*, 2013.

[19] M. Egele, C. Kruegel, E. Kirda, and G. Vigna, "PiOS: Detecting privacy leaks in iOS applications," in *Proc. of NDSS*, 2011.

[20] S. Arzt, S. Rasthofer, C. Fritz, E. Bodden, A. Bartel, J. Klein, Y. Le Traon, D. Octeau, and P. McDaniel, "FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps," in *Proc. of PLDI*, 2014.

[21] A. Razaghpanah, N. Vallina-Rodriguez, S. Sundaresan, C. Kreibich, P. Gill, M. Allman, and V. Paxson, "Haystack: In Situ Mobile Traffic Analysis in User Space," *ArXiv e-prints*, 2015.

[22] S. Zimmeck, J. S. Li, H. Kim, S. M. Bellovin, and T. Jebara, "A privacy analysis of cross-device tracking," in *Proc. of USENIX Security*, 2017.

[23] J. Brookman, P. Rouge, A. Alva, and C. Yeung, "Cross-device tracking: Measurement and disclosures," *Proc. of PETS*, 2017.

[24] The New York Times, "Unroll.me Service Faces Backlash Over a Widespread Practice: Selling User Data," https:

//www.nytimes.com/2017/04/24/technology/personal-data-firm-slice-unroll-me-backlash-uber.html.

[25] Dare Obasanjo, "How Facebook Knows What You Looked at on Amazon," http://www.25hoursaday.com/weblog/2014/02/17/HowFacebookKnowsWhatYouLookedAtOnAmazon.aspx.

[26] Reuters, "Yahoo says one billion accounts exposed in newly discovered security breach," http://www.reuters.com/article/us-yahoo-cyber-idUSKBN1432WZ.

[27] Recode, "Yahoo is expected to confirm a massive data breach, impacting hundreds of millions of users," https://www.recode.net/2016/9/22/13012836/.

[28] "Children's Online Privacy Protection Rule ("COPPA")," https://www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/childrens-online-privacy-protection-rule.

[29] "Mobile Advertising Network InMobi Settles FTC Charges," https://www.ftc.gov/news-events/press-releases/2016/06/mobile-advertising-network-inmobi-settles-ftc-charges-it-tracked.

[30] "Lawsuit claims Disney illegally collected data in kids apps," //www.engadget.com/2017/08/09/disney-illegally-collected-data-kids-apps/.

[31] "EU General Data Protection Regulation," http://data.consilium.europa.eu/doc/document/ST-9565-2015-INIT/en/pdf.

[32] M. Lindorfer, M. Neugschwandtner, L. Weichselbaum, Y. Fratantonio, V. Van Der Veen, and C. Platzer, "Andrubis–1,000,000 apps later: A view on current android malware behaviors," in *Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS), 2014 Third International Workshop on*. IEEE, 2014.

[33] Facebook for Developers, "The Graph API," https://developers.facebook.com/docs/graph-api.

[34] A. Le, J. Varmarken, S. Langhoff, A. Shuba, M. Gjoka, and A. Markopolou, "AntMonitor: A System for Monitoring from Mobile Devices," in *Proc. of ACM C2B1D*, 2015.

[35] Y. Song and U. Hengartner, "PrivacyGuard: A VPN-based Platform to Detect Information Leakage on Android Devices," in *Proc. of ACM CCS SPSM Workshop*, 2015.

[36] V. Paxson, "Bro: a system for detecting network intruders in real-time," *Computer networks*, 1999.

[37] Suricata, "Open Source IDS / IPS / NSM engine," https://suricata-ids.org.

[38] "Snort," https://www.snort.org.

[39] D. Dittrich and E. Kenneally, "The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research," *US DHS*, 2012.

[40] "MoboGenie," http://www.mobogenie.com/.

[41] "Aptoide. Your Android App Store." https://www.aptoide.com.

[42] "Free WhatsDog," com.newmesseg.freewhats2016.

[43] "EasyList Overview," https://easylist.to/.

[44] MalwareBytes, "hpHosts," http://hosts-file.net/.

[45] Intel Security/McAfee, "Customer URL Ticketing System," http://www.trustedsource.org/.

[46] "OpenDNS Domain Tagging," https://domain.opendns.com.

[47] "VirusTotal," https://www.virustotal.com.

[48] "Alexa top sites," http://www.alexa.com/.

[49] "DuckDuckGo Search Engine," http://www.duckduckgo.com/.

[50] "HashingVectorizer," http://scikit-learn.org/stable/modules/generated/.

[51] "ADB Shell. Getprop," http://adbshell.com/commands/adb-shell-getprop.

[52] Android Developer's Documentation, "Best Practices for Unique Identifiers," https://developer.android.com/training/articles/user-data-ids.html.

[53] Google. Android Developer Policy Center, "Usage of Android Advertising ID," https://play.google.com/about/monetization-ads/ads/ad-id/.

[54] "Crunchbase," https://www.crunchbase.com/.

[55] "D&B Hoovers," http://www.hoovers.com.

[56] "Maxmind. GeoLite2 Database," http://dev.maxmind.com/geoip/geoip2/geolite2/.

[57] "GDPR Portal: Site Overview," https://www.eugdpr.org.

[58] "Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications) ," http://eur-lex.europa.eu.

[59] ESRB, "ESRB Ratings Guide," http://www.esrb.org/ratings/ratings_guide.aspx.

[60] R. Nithyanand, S. Khattak, M. Javed, N. Vallina-Rodriguez, M. Falahrastegar, J. E. Powles, E. De Cristofaro, H. Haddadi, and S. J. Murdoch, "Ad-blocking and counter blocking: A slice of the arms race," in *Proc. of USENIX FOCI*, 2016.

[61] S. Zimmeck and S. M. Bellovin, "Privee: An architecture for automatically analyzing web privacy policies," in *Proc. of USENIX Security*, 2014.

[62] Y. Zhang, M. Yang, B. Xu, Z. Yang, G. Gu, P. Ning, X. Wang, and B. Zang, "Vetting undesirable behaviors in android apps with permission use analysis," in *Proc. of ACM CCS*, 2013.

[63] P. Hornyack, S. Han, J. Jung, S. Schechter, and D. Wetherall, "These aren't the droids you're looking for: Retrofitting android to protect data from imperious applications," in *Proc. of ACM CCS*, 2011.

[64] Android Developer's Documentation, "Android developers: UI/application exerciser monkey," http://developer.android.com/tools/help/monkey.html.

[65] C. Hu and I. Neamtiu, "Automating GUI testing for android applications," in *Proc. of ACM AST*, 2011.

[66] T. Takala, M. Katara, and J. Harty, "Experiences of system-level model-based gui testing of an android application," in *IEEE International Conference on Software Testing, Verification and Validation*, 2011.

[67] K. W. Y. Au, Y. F. Zhou, Z. Huang, and D. Lie, "PScout: Analyzing Android permission specification," in *Proc. of ACM CCS*, 2012.

[68] I. Leontiadis, C. Efstratiou, M. Picone, and C. Mascolo, "Don't kill my ads!: balancing privacy in an ad-supported mobile application market," in *Proc. of ACM HotMobile*, 2012.

[69] A. Continella, Y. Fratantonio, M. Lindorfer, A. Puccetti, A. Zand, C. Kruegel, and G. Vigna, "Obfuscation-resilient privacy leak detection for mobile apps through differential analysis," in *Proc. of NDSS*, 2017.

[70] M. Backes, S. Bugiel, and E. Derr, "Reliable third-party library detection in android and its security applications," in *Proc. of ACM CCS*, 2016.