In-Depth Survey of Digital Advertising Technologies

Gong Chen, *Student Member, IEEE*, Jacob H. Cox, Jr., *Student Member, IEEE*,
A. Selcuk Uluagac, *Senior Member, IEEE*, and John A. Copeland, *Life Fellow, IEEE*

***Abstract*—Some of the world's most well-known IT companies are in fact advertising companies deriving their primary revenues through digital advertising. For this reason, these IT giants are able to continually drive the evolutions of information technology in ways that serve to enhance our everyday lives. The benefits of this relationship include free web browsers with powerful search engines and mobile applications. Still, it turns out that "free" comes at a cost that is paid through our interactions within a digital advertising ecosystem. Digital advertising is not without its challenges. Issues originate from the complex platforms utilized to support advertising over web and mobile application interfaces. This is especially true for advertising links. Additionally, as new methods for advertising develop so too does the potential for impacting its underlying ecosystem for good or ill. Accordingly, researchers are interested in understanding this ecosystem, the factors that impact it, and the strategies for improving it. The major contribution of this survey is that it is the first review of the digital advertising ecosystem as it applies to online websites and mobile applications. In doing so, we explain the digital advertising relationships within this ecosystem along with their technical, social, political, and physical implications. Furthermore, advertising principles along with a variation of other advertising approaches, both legitimate and malicious, are explored in order to compare and contrast competing digital advertising methods.**

***Index Terms*—Digital advertising, mobile advertising, online advertising, monetization, networking, privacy, security, advertising networks, advertising policy.**

I. Introduction

DIGITAL advertising (a.k.a., Internet Advertising), including online advertising and mobile advertising, is a driving force for monetization throughout the Internet. While online advertising has continued to change the rules within the industry since the 1980's, a relative new comer, in-app mobile advertising, has taken hold in the cell phone market. Together, these platforms have driven advertising revenues to all-time highs. According to the Interactive Advertising Bureau's (IAB) 2014 first and third quarter reports, Internet advertising achieved year-over-year revenue increases of 17% [1], [2]. These reports also specify quarterly earnings of $11.6B and $12.4B, respectively. While growth in Internet advertising is impressive, global mobile advertising revenue achieved a tremendous 92% growth from 2012 ($10B) to 2013 ($19.3B) . Gartner further predicts an additional increase of 100% to occur between 2013 and 2016. This high potential for generating revenue makes digital advertising a major driving factor for web and mobile technology.

For instance, consumers are now exposed to HTML5 rich media advertisements ("ads" for brevity) that are displayed directly to their devices. Keeping pace with this technology, cellular networks continue to upgrade their infrastructure to support the extra overhead. This, of course, ultimately improves the customer experience while browsing websites or using mobile applications ("apps" for brevity). Additionally, advertising technology is driving political legislation that is designed to protect user privacy. For example, a 2013 report [3] presented by both the Federal Trade Commission (FTC) and the California's Attorney General calls for limiting the cross-application tracking of users by constraining the use of data collection and storage by app developers. The same report also recommends eliminating access to global device identifiers (i.e., the identification number for a mobile device) by app developers. Likewise, developers are called to encrypt data, limit access to user information, and use special notices to call the user's attention to the app's privacy settings. Other privacy practices, such as explaining how a user's personally identifiable information (PII) is collected and used, have already been mandated in some US states and Europe [3].

From a high-level perspective, *advertising brokers* essentially work with *advertisers* to place their ads on websites and mobile apps that are willing to host them. The forms of mobile ads may include mobile video and TV, text and multimedia messaging, mobile web, and numerous applications. Our list is more exclusive. Given the rising importance of mobile ads, in this survey, we primarily focus on online ads and in-app ads displayed on mobile devices. Additionally, in order to provide a survey that is both timely and relevant to the state of mobile advertising, we constrain the scope of this survey to technical papers published from 2007 through 2016 and recent articles from related websites. Moreover, we compare and contrast in-app mobile ads with web-based ads to reveal the underlying statistics that may be of interest to mobile advertising researchers, professionals, and tech-savvy hobbyists. Numerous terms are also introduced in this survey, and they are listed in Table I for the reader's benefit.

This survey makes the following contributions:

- This is the first technical survey paper relating to both online and mobile advertising.
- It explores algorithms relevant to digital advertising.
- It contrasts online and mobile advertising roles.
- It compares objectives of advertising research and development in academia and industry.

The rest of this survey, as depicted in Figure 1, is structured as follows. Section II provides the background for

Manuscript received June 17, 2015; revised November 24, 2015; accepted December 26, 2015. Date of publication January 25, 2016; date of current version August 19, 2016.

G. Chen, J. H. Cox, and J. A. Copeland are with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: gong.chen@gatech.edu).

A. S. Uluagac is with the Department of Electrical and Computer Engineering, Florida International University, Miami, FL 33199 USA.

Digital Object Identifier 10.1109/COMST.2016.2519912

TABLE I
NOMENCLATURE IN DIGITAL ADVERTISING

Term	Meaning
ad campaign	A series of advertising messages that promote a single theme or message
ad creative	An object that contains all the data for visually rendering the ad itself
ad impression	An instance that a user is exposed to an ad
ad space	An area within a website or an app dedicated to displayed ads
ad unit	An advertising vehicle that includes creative assets inside creative assets inside an ad space
landing page	A web page specifically designed for people visiting via a pre-determined path
remnant inventory	An advertising space that a publisher is unable to sell directly through its sales force
ad network	A system that aggregates ad inventory from web publishers and app developers to efficiently match the inventory with advertisers' demands
ad exchange	A technology platform that facilitates the buying and selling of remnant ad inventory that can be bid on in real time
advertiser	An organization that wants to get its message to the right audience, efficiently and effectively
publisher	An organization looking to maximize the monetization of its space
DSP	"Demand-Side Platform", a platform that enables advertisers to manage all ad exchanges through a single interface
SSP	"Supply-Side Platform", a platform that enables publishers to manage and sell its ad inventory through a single interface
CPM/PPM	"Cost/Pay per Mille or thousand impressions", advertisers pay for ad impressions in bulk
CPC/PPC	"Cost/Pay per Click", advertisers pay only if a user clicks on their ads
CPL/PPL	"Cost/Pay per Lead", advertisers only pay for an interested lead (i.e., a qualified sign-up, or a "like" in case of Facebook) regardless of how many impressions or clicks their ad received
CPA/PPA	"Cost/Pay per Action", advertisers pay only if a user conducts a closed sale or a particular action at the moment
CTR	"Click-Through Rate", a measurement of how many users clicked on an ad
conversion rate	A measurement of how many users take action beyond viewing or interacting with an ad
eCPM	"effective CPM", a revenue model to determine the effective cost per thousand impressions
RTB	"Real-Time Bidding", a means by which advertising inventory is bought and sold on a per-impression basis, via programmatic instantaneous auction, similar to financial markets

digital advertising and includes basic concepts and underlying statistics. In Section III, we identify the primary monetization methods utilized in digital advertising. Section IV covers the networking schemes inherent in advertising ecosystems. Section V addresses the privacy concerns of users and the goals of advertisers. Section VI analyzes the various malware and countermeasures impacting digital advertising today. The human factors that steer advertising research are then covered in section VII. Section VIII highlights the energy and data constraints of mobile devices and their relevance to online and mobile advertising. Finally, in section IX, we discuss additional points of interest from the survey and then conclude in section X.

II. BACKGROUND

Digital advertising is a relatively new form of marketing that is quickly proliferating throughout the Internet. As online

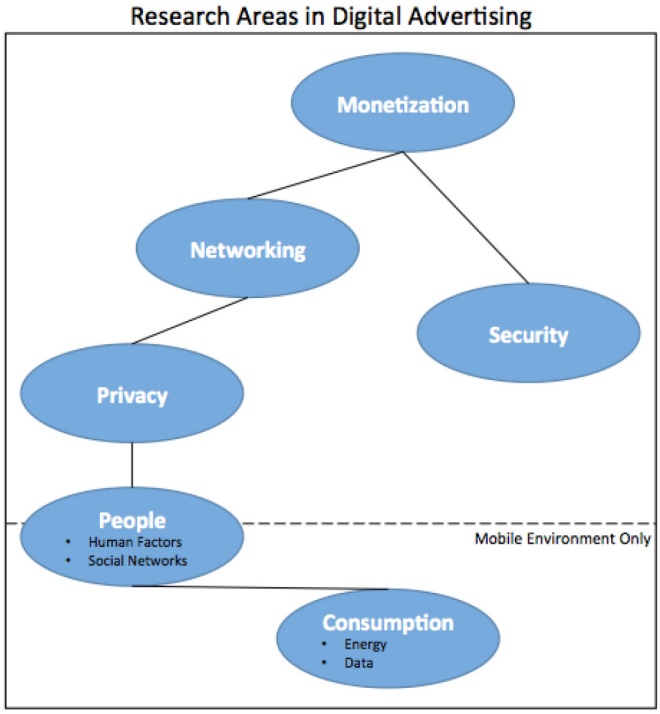


Fig. 1. Paper Structure.

TABLE II
TIMELINE OF ONLINE ADVERTISING

1993	Global Network Navigator became 1st publisher to sell banner ads
1994	AT&T became one of the 1st advertisers to buy banner ads
1995	Yahoo launched the 1st keyword ad ("Golf")
1996	The newly launched company, DoubleClick (by Google since 2007), developed DART[1] to price advertisers based on CPM
1996	HP's "The pong banner" became the 1st rich media banner ad
1998	CPC model became a means to help search engines generate revenue (i.e., keyword auction)
2000	Google AdWords was launched
2005	RightMedia appeared as the 1st ad exchange
2006	Facebook Ads was launched
2007	YouTube (by Google since 2006) rolled out InVideo Ads
2009	Real-time bidding was adopted by many ad exchanges (e.g., DoubleClick)
2010	Promoted Tweets and then Promoted Trends were introduced by Twitter

and mobile platforms offer marketers more venues to gain customers, these platforms have also experienced numerous evolutions in technology, policy, and security. For historical context, Table II and Table III offer timelines depicting notable milestones in online advertising and mobile advertising respectively.

Digital advertising consists of an infrastructure that allows *users*, *publishers*[1] and *advertisers* to interact in an advertising ecosystem with brokers. While publishers provide ad spaces on their websites, advertisers spend money to have their ads placed on those sites. Thus, collaboration between publishers

[1]While the term *publisher* is used in online advertising, the term *developer* is used in mobile advertising.

TABLE III
TIMELINE OF MOBILE ADVERTISING

2G era	
2000	SMS advertising was launched in Finland
2.5G/GPRS era	
2003	MMS advertising was tested in SmartRotuaari
3G era	
2006	AdMob (by Google since 2009) was launched
2007	Mobile web advertising started along with smartphone
2007	The 1st mobile ad exchange created
2007	AdSense for Mobile was launched
2009	In-app advertising, mobile video advertising & QR code advertising started
4G LTE era	
2010	Apple iAd was launched
2012	Facebook Ads on Mobile was launched

and advertisers is required. While it is possible for a few large advertisers to directly negotiate with a few large publishers as has been done with traditional billboards and paper media, this direct collaboration is not suitable for the vast number of publishers and advertisers on the Internet today. Brokers, including *advertising networks* ("ad networks" for brevity) and *advertising exchanges* ("ad exchanges" for brevity), represent multiple agents. While ad networks basically intermediate between publishers and advertisers, ad exchanges provide auctions for ad spaces among various entities, which include ad networks, supply-side platforms, and demand-side platforms. Therefore, when a user views multiple ads on a web page, these *ad impressions* may originate from multiple sources, as depicted in Figure 2.

A. Ecosystem for Online Advertising

The red highlighted path in Figure 2 represents a simple model widely used in academia. In this figure, the *advertiser* represents the money source, while the *user* serves as the initiator of actions. However, the user must initiate all actions with the publisher (or application provided by the app developer). The *ad network* is the entity that links all together. Likewise, the red lines represent the simplest path with fewest hops between the advertiser and the user. However, the advertising ecosystem is not that simple. Other entities also exist within this model as indicated in the multiple other arrows showing various interactions. For instance, the *DSP* serves to aggregate multiple advertisers, and the *SSP* aggregates multiple publishers. Additionally, the *ad exchange* is similar to the NYSE in that it gathers all buyers and sellers together. These entities further breakdown into buyers and sellers. Buyers include publishers, SSPs, and ad networks, while sellers include advertisers, DSPs, and other ad networks. Note that ad networks and ad exchanges assume both roles. Also, within this hybrid ecosystem, all paths from the advertiser to users are possible. Money flows are shown with single arrows while information flow is depicted with double arrows.

Admittedly, an ad network usually manages publishers and advertisers separately with different servers (i.e., publisher ad server and advertiser ad server); however, these servers are considered as one for the purpose of this survey. There are also two types of publishers, which include 1) the publishers owned and operated by the ad networks (e.g., Google Search or Facebook) and 2) the syndicated publishers not managed by the ad networks (e.g., reddit or TechCrunch) [4]. These syndicated publishers are either premium publishers, bound by their service level agreement (SLA), or self-serve publishers, allowed to sign up with ad networks at any time. We focus on the more prevalent case, which includes self-serve publishers. With these publishers, an advertising ID and appropriate source code must be obtained from an ad network and placed in the publisher's source code in order to display the ad impressions on the publisher's website.

Having explained how ad impressions are incorporated with a self-serve publisher's website, we now explore the process that occurs whenever a potential customer clicks on an ad impression. In Figure 3, the steps of this process are depicted. When a user starts loading a web page (steps 1 and 2), ad impressions are requested by algorithms existing in the web page's source code from the ad network's server (step 3). After identifying the publisher's advertising ID, which is assigned by the ad network, the ad network's server logs the request (step 4), applies the rules previously established with its advertisers, and returns an ad that includes a unique identifier for click tracking (step 5). Once a user clicks on the ad (step 6), an HTTP GET request is sent to the ad network (step 7). This is considered a click-through event by the ad network, and it is logged for billing purposes (step 8). The ad network then redirects the browser to the advertiser's landing page through the client via HTTP 302 status code (step 9). The landing page is hosted on either the advertiser's ad server directly or on a content delivery network (CDN)[2]. Having reached this point, the user is now able to browse items on the landing page (steps 10 and 11). According to Wang et al. [5], the content delivery for online advertising may take about 100 ms for distributed network architecture (e.g., AOL/Akamai: 87 ms and Google: 122 ms) or around 200 ms for a standalone server (e.g., Adblade: 207 ms).

Usually, when an ad network first encounters a user, it sends a cookie to the user or uses other indirect methods (e.g., an IP address and HTTP user agent combination). The ad network is then later able to complete either an API call or an HTTP response with the user. Resultantly, ad networks are able to label web users, determine their browsing patterns, and measure the effectiveness of specific ad campaigns.

Any empty ad space must be sold to an advertiser within a limited amount of time (e.g., 100 ms for DoubleClick Ad Exchange); otherwise, a blank space is left on the web page and a loss of revenue occurs. In order to maximize potential revenue for the whole ecosystem, multiple ad networks formalized partnerships that included predefined revenue share. Whenever needed, an ad network will demand a filling from each of the partners. This method, however, is still highly inefficient. Thus, the ad exchange (e.g., Google's DoubleClick and Yahoo!'s RightMedia) emerged to solve this problem by loosely aggregating all platforms together. Based on Real-Time Bidding (RTB), the ad exchange is able to perform massive

[2]CDNs (e.g., Akamai) provide web content via their servers around the world in order to decrease latency.

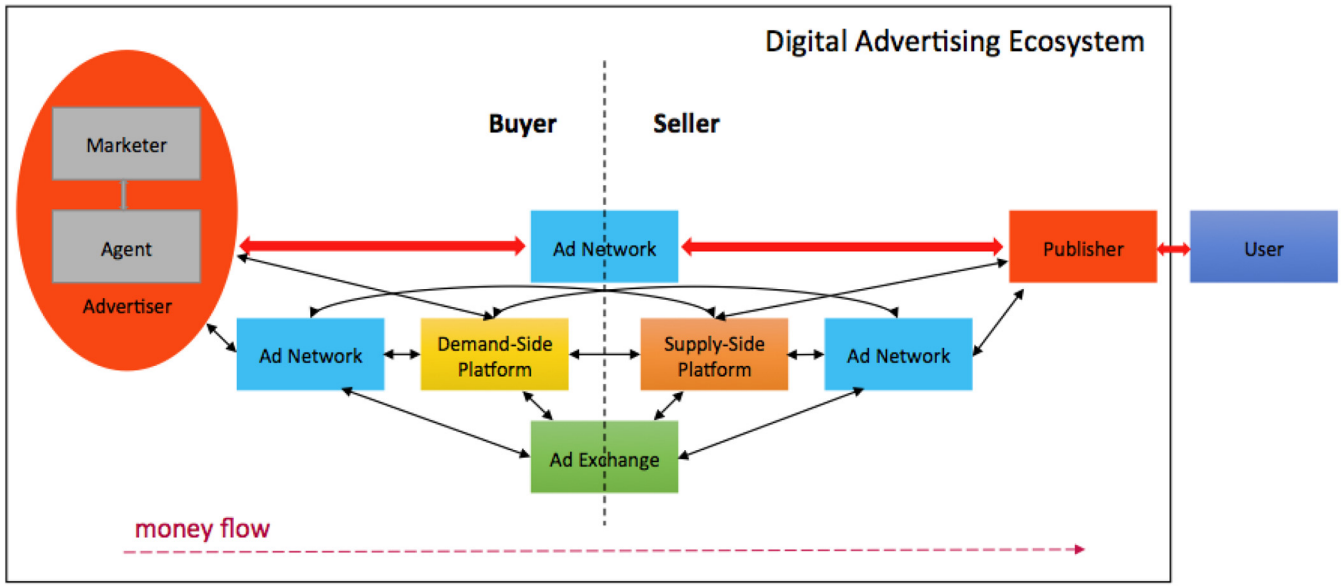


Fig. 2. Digital Advertising Ecosystem.

Fig. 3. Ad flow.

aggregation and enable single ad space bidding, which results in higher revenues for publishers and ad networks, while including fine-grained ad targeting for advertisers. In order to fulfill an ad request, users must initiate at least four sequential requests as shown in Figure 3: first to the publisher's ad server, second to the ad exchange, third to the winning advertiser's ad server, and finally to the CDN.

Along with the development of ad exchanges, new problems ranging from auction to optimization have been raised [6]. Problems, include fixed-percentage sharing schemes and fierce competition from comparable market places [7]. Yet, in practice, the concerns of an ad exchange are still quite similar to those found in ad networks. It may be that these problems have yet to be fully identified since there are still few large-scale datasets for ad exchanges. For example, by using the ad logs of an ad network in RightMedia, Stone-Gross et al. [8] are able to locate suspicious publishers, mainly by using local traffic and individual detection features. Still, they are unable to detect the overall fraudulent activities across the platform.

On a larger scale, RTB is based on the second price auction model. As a result, the publisher receives compensation equivalent to the second-highest bid to be paid by the winning advertiser. To support this phase and verify the integrity of auctions, VEX [9] requires that each of the bidding advertisers initiate a three-way communication, within the ad exchange. While doing so, VEX is still able to achieve reasonable latency and processing costs.

B. *Ecosystem for Mobile Advertising*

According to [10], in contrast to online advertising where ad networks employ client redirection, *server redirection* is used in mobile ecosystems to accommodate the slower connectivity and limited data usage. Also, unlike online advertising, mobile ads meld with applications on mobile devices. As a result, this form of advertising can offer advertisers unique ways to interact with users and their personal information. We now discuss the platforms and libraries available to mobile app designers seeking to monetize their apps through in-app ads.

1) Mobile Platforms: According to Gartner [11], Android and iOS are the two most prevalent smartphone platforms in the world. While both Android and iOS platforms attract third-party developers to create mobile apps, all of these apps must run as a non-privileged user in a sandboxed app execution environment. Also, developers may encounter limited memory for their apps across all mobile platforms. For example, in a Windows Phone, apps are allowed to consume no

more than 90MB of RAM at runtime [12]. Furthermore, smartphones contain sensors (e.g., GPS, accelerometer, gyroscope, magnetometer, luxmeter, microphone, camera, and speaker). Therefore, permission systems are employed to control access to system resources via application programming interface (API) calls.

Permissions for apps can be granted in one of two ways: 1) Time-of-Use and 2) Install-Time. The former, used in iOS and also in web browsers, asks users to approve or deny permissions for resources (e.g., location) whenever the app triggers a privileged API call. The latter, used in Android, prompts users to approve *dangerous permissions* listed in a manifest file before installing the app. These dangerous permissions (e.g., RECORD_AUDIO) can either access private user data or exercise control over the device. Regrettably, permission authorization during installation is all or nothing. This means that a user must either grant access for the requested user-owned resources during installation or terminate the install.

Android apps lack a single entry point, like main(), and consist of one or more app components (i.e., activity, service, content provider, and broadcast receiver) through which they interact with the system [13]. Permission re-delegation can also happen with components running transparently in the background when a privileged app makes an API call [14]. For each app, a unique Android UID is associated with its permissions list (specified in the app's AndroidManifest.xml file). These system permissions can be categorized into 4 protection levels (i.e., normal, dangerous, signature, signatureOrSystem) [15]. Usually, app developers are required to obey *the principle of least privilege* during app development. However, Stowaway [16] still identifies the overprivilege issues (i.e., demand unnecessary permissions) in Android apps. As a counter, PScout [17] uses static analysis to reveal the unnecessary use of non-system permissions for a few cases, and Quire [18] allows an app to adjust its privilege by tracking inter-process communication (IPC) call chains.

As for iOS, third-party apps are only allowed to use the public frameworks. Any use of a private API results in a direct rejection during Apple's vetting process, which mandates that all executable code be signed with an Apple-issued certificate. The vetting process is alleged to be stringently safe, although attackers can still realize malicious activities by constructing new execution paths after the app review process [19] or inject malware via a malicious charger [20].

2) Advertising Libraries: Pioneering discussions for mobile advertising in Android and iOS may be attributed to TaintDroid [21] and PiOS [22], respectively. TaintDroid finds that half of its studied apps share location information with third-party ad servers in plaintext or in binary format without user consent. Likewise, from a study of 1.1K top-selling free Android apps, Enck et al. [23] confirm that advertising & analytics ("A&A" for brevity) services are willing to probe permissions and acquire other critical information such as the International Mobile Station Equipment Identity (IMEI) of a user's mobile phone. As for iOS, PiOS points out that more than half of the studied apps result in leaking unique device identifiers (UDID) because of its embedded A&A libraries. Although Egele et al. [22] only predict that the 40-byte identifiers collected by AdMob are linked with Google accounts, Smith [24] confirms that a number of apps are collecting UDID and login information for Amazon, Facebook and/or Twitter, and they are able to track the users in real time, revealing their usage patterns. When combined with long-lived tracking cookies or time-stamped IP addresses, users' location information and other private data are surrendered and registered. In a follow-up study, Smith [25] also claims that millions of UDIDs are leaked to app developers and even hackers. Likewise, Smith [25] states that iAd banners also collect and transfer UDIDs to Apple.

TABLE IV
MOST POPULAR AD STYLES IN MOBILE APPS

Ad Style	Position	Used by	CPM [36]
Banner Ads	Full-width ads placing at the top or bottom of the screen	Mobile web libraries	\$0.02-\$0.10
Interstitial Ads	Full-screen ads placing before or after a content page	Rich media libraries	\$0.5-\$10

In order to get continuous revenue from apps, many mobile app developers consider mobile ads. For example, Flurry works independently from Angry Birds to collect and upload statistics to a remote server in order to download and render ads during gameplay [26]. The developer registers their financial information [27] and their app's metadata, which includes app name, category, requested permissions, app store link, average rating, and number of downloads [12]. In turn, the developer receives an identifier and a Software Development Kit (SDK). The SDK provides the developer with the necessary instructions for using its advertising library ("ad library" for brevity), and it provides abstraction for the complex fetching and reporting protocols required by ad networks. Accordingly, the libraries may use *WebView* to facilitate ad loading; however, the use of JavaScript interfaces in WebView may lead to attacks from malicious web pages or other apps [28].

There are many other vulnerabilities to user privacy as well [29]. Among the papers we surveyed, the number of ad libraries implanted in each app are reported with divergent results from one researcher to the next. While most apps bundle more than one ad library [23], [30], researchers [31] claim that over 75% of 32K apps bundle only one ad library with approximately 1% bundling more than five ad libraries. Using ad libraries, ad servers are able to track click events via a unique click URL associated with each ad impression to appropriately distribute ad revenues [32]. In Android, developers are asked to include the permissions required by the ad library within the Android Package Kit (APK). Since all permissions used by an app are stored in the AndroidManifest.xml file, an ad network is able to access the same services as the host app. Each ad library may require different permissions, but the INTERNET permission is always required [33].

Mobile ads are used to advertise not only external brands, but also apps in the marketplace [34], and they are commonly implanted into banners and interstitials [35] as listed in Table IV. Static ads may be at most 15KB in size; however, the average size of these ads falls between 1.5KB and 5KB for uploading and downloading, respectively [12]. Besides banners and interstitials, there are also notification ads (e.g., AirPush,

MobPartner, and SendDroid) and icon ads (e.g., EverBadge and AirPush) in Android. As the names imply, one is displayed in the notification area of an app while the other is placed as a shortcut on the app's home screen. Of course, both of these ad methods are unpopular for understandable reasons. Primarily, they violate Google's Developer Content Policy. Secondly, they negatively impact user satisfaction. As for ad libraries, they can be classified into three groups [27]: 1) mobile web library, 2) rich media[3] library, and 3) ad mediator[4]. While the latter group uses traditional web technologies to request, deliver, and display banner and interstitial ads, the previous two groups employ HTML5 to create video ads. Since mobile app developers may bundle more than one ad library, ad mediators (e.g., AdWhirl, MobClix, and Burstly [35]) have emerged to coordinate various ad libraries in an app. Furthermore, they query for ads from the subscribed ad networks based on factors such as developer's ad percentage distribution. Nowadays, most ad networks provide mediation options. For example, AdMob allows mediation with 16 other ad networks (e.g., Adfonic, Flurry, InMobi, iAd, Jumptap, Millennial Media, etc.). The introduction of these mediation services potentially increase developer's income and in cases where one ad network fails to render an ad, another is able to fill the space.

The popularity of ad libraries can be calculated by either the number of apps using the library or the number of app installations including the library [33]. While the top 10, top 25, and top 33 ad libraries respectively represent 71%, 90%, and 95% of market share based on app installations [33], AdMob, like other Google Ads & Analytics services, owns a dominant position in Android. In 2010, AdMob served ads for more than 15K mobile websites and applications around the world [37]. After studying 55K free apps from Google Play, Tongaonkar et al. [38] reveal that nearly 12K apps bundle the most popular ad library (i.e., Google Ads). Similarly, Grace et al. [27] reveal that, in Android apps, Google's AdMob, AdSense, and Analytics networks are listed in the top five of 100 comparable libraries. Flurry and Millennial Media are the other two libraries making the top five list.

In [35], during a one-day network traffic trace of a major European mobile carrier, approximately 4.5% of Android and iOS users were discovered to click on an AdMob ad at least once. Presumably, over two-thirds of ad traffic from Android and iOS corresponds to AdMob. Even more prevalent is AdMob on iPad devices. On these devices, AdMob surpasses the Apple-owned ad library (i.e., iAd) with over 90% of market share [35]. Because of its prevalence, AdMob is frequently used for case study examples by researchers. After importing AdMob, the developer deals with the layout by either declaring an AdView in the XML layout or inserting an AdView instance directly into the view hierarchy [39]. In order to request ad content, the ad library triggers an AdRequest and loads the ad in the AdView [40]. Like other ad libraries (e.g., AirPush), AdMob obfuscates the API calls to protect the intellectual property [27], [40]. Yet, it also requests that separate and explicit GPS coordinates be provided [29]. These mobile advertising interactions are mostly accomplished through HTTP GET requests in plaintext using REST APIs [35]. Upon receiving the request, AdMob mediates internally all Google ad services (e.g., DoubleClick and AdSense). Of course, other ad networks may utilize different protocols. For example, InMobi sends a single HTTP POST request per ad, and Millennial Media establishes two HTTP connections in order to get the ad and associated static content from different servers [35].

[3]These include a broad range of interactive and engaging ad formats, including expandable banners and embedded audio/video ads.

[4]Platform which allows developers to strategically sell remnant and unsold inventory by using multiple ad networks.

TABLE V
RELATION BETWEEN AD TYPES & PRICING MODELS

Display Advertising (context: all cases, user data: bidding only)	CPM
Search Advertising (keywords)	CPC/CPA/CPL

III. MONETIZATION

Generally, there are two ad types (i.e., display advertising and search advertising) and four pricing models (i.e., CPM, CPC, CPA, and CPL) in digital advertising. The relation between ad types and pricing models is depicted in Table V. Note that the unaligned row is intended to show that display advertising also includes CPC/CPA/CPL. Display advertising also includes contextual advertising and behavioral advertising. Additionally, of these models depicted, CPM and CPC are the two most common.

For budgeting purposes, many advertisers allocate a budget for their ad (or ads) to span a specified time period. For example, category-based ads (e.g., contextual display ads) are renewed on a monthly basis (e.g., 5K clicks/month on Revisitors.com costs $28.95), whereas keyword-based ads (e.g., search ads) are paid daily (e.g., Google: $4.5/click & Yahoo!: $3.0/click) [41]. Publishers likewise seek profits and receive up to 70% of what is paid by the ads' advertisers. Lastly, ad networks receive the remaining share of an advertiser's payment. For example, ad networks keep around 30% of the advertiser's payment (e.g., 32% for Google AdSense) and distributes the remaining portion to its publishers. In other cases, publishers must provide a set number of conversions in order to collect from ad networks or advertisers (e.g., smart pricing). Ad networks, however, may use the ad syndication business model to play arbitrage (i.e., buy cheap ad spaces and resell them at a higher cost) [8]. For example, Criteo mediates between publishers and advertisers by paying CPM from publishers and selling CPC/CPA to advertisers [42]. Finally, eCPM metrics are used to evaluate conversion rates and overall success.

A. Pricing Mechanisms for Online Ads

1) CPM Revenue Model: CPM is a relative easy pricing model. In CPM, the cost of impressions ($CPM(u, p, a)$) depends on the product of three variables [43], run of network (RON_a), traffic quality multiplier (TQM_p), and user intent ($I_a(u)$): ($CPM(u, p, a) = RON_a \times TQM_p \times I_a(u)$.

TABLE VI
FEATURED AUCTION-RELATED ISSUES IN DIGITAL ADVERTISING

Project	Feature	Description
[47]	RTB, display advertising, logistic regression	Use logistic regression with the iPinYou RTB dataset to learn and predict CTR, and find a non-linear concave relationship between bid and CTR
[48]	RTB, search advertising, least mean square regression, hierarchical Laplace smoothing	Use actual bids with penalized weighted least mean square regression to predict advanced match after learning conversion data with hierarchical Laplace smoothing, and find the bid data is more useful for inferring the appropriate bid value than the conversion data
[42]	CPA, display advertising, conversion prediction, EM algorithm, L-BFGS optimizer	Use the modified Expectation-Maximization algorithm (at the M step, use L-BFGS to optimize the negative log-likelihood by gradient descent) to demonstrate the efficiency and effectiveness of the proposed delayed feedback model
[49]	search advertising, click prediction, FTRL-Proximal online learning algorithm	Improve the traditional supervised learning based on FTRL-Proximal online learning algorithm on Google's deployed CTR prediction system
[50]	display advertising, revenue maximization	Use iterative optimization algorithm with prices adjusted to maximize revenue, and estimate the gradient of impression allocations as regards inventory prices
[51]	budget optimization	Devise an optimal (1-1/e)-approximation algorithm for the source-side model, and demonstrate the difficulties of using the target-side model to maximize influence than the source-side model

The normal RON campaign in ad network a values the base price (\$1.98 on average) for advertisers because it is not specifically targeted to users [44]. TQM is determined by the potential value of an ad space for publisher p. For example, the ad spaces of a popular publisher may warrant a higher rate (2 for popular publishers, 0.1 for disreputable sites, and 1 for the remaining publishers). Also, the location of an ad space matters if it is to fall within a typical user's web page viewing habits [45]. Additionally, user intent is inferred through web tracking of user u (ranges from 2 to 10, but with an average of 3.3 [46]). Similarly, a study of the CPM model and HTTP datasets reveals that the intent for 50% of users can be accurately estimated [43].

2) Auctions: The majority of ad spaces for search ads and behavioral display ads are sold via RTB. However, multiple methods exist for RTB. Some of which are listed in Table VI along with their descriptions. Other auction features include CPA, learning algorithms, and budget optimization.

Various auction mechanisms also exist for single ad or multiple ad bidding. Three different auction types are discussed by Stavrogiannis et al. [52] and include first price sealed bid auction and pre and post award Vickrey auctions. They find that the pre-award Vickrey auction is the least efficient of these models. They also find, in cases of simultaneous, multiple item bidding, that the majority of evaluated companies preferred either generalized second price (GSP) model or the Vickrey-Clarke-Groves (VCG) model [53]. For instance, Google, Bing, and Yahoo! use GSP while Facebook uses VCG [48]. The performance of these models is further evaluated in work by Edelman et al. [54]. They find that both of these models calculate the advertiser's CPC according to the mathematical product of their bid and quality score. When a user queries the Google search engine, the search engine results page (SERP) renders both relevant query results (i.e., those closely matching the user's search query) and sponsored results (i.e., search ads) [54]. These ads are then assigned based on an auction among all advertisers who bid for the ad impressions. To participate in the auction, the advertisers submit their ad tags, which are the bids linked to each ad, and their targeting information, consisting of a keyword and location. Since a search engine may contain millions of bidding keywords [12], some researchers [55] have investigated how better ad indexing can yield faster ad retrieval. Consequently, when the ad network receives an ad request, an auction is performed among the ads matching the user's search criteria and the appropriate ads are displayed. As a result, the first few ads appear in Google's search results in accordance with their ranking as determined by their highest expected revenue or ranking score (i.e., bid * quality score). But, the cost for a clicked search ad is then calculated based on the next lower ranked ad, as demonstrated in Table VII. Optimizing GSP is further discussed in [56]–[58]. However, the VCG auction system charges the winning bidder of an item the potential loss of all other bidders caused by the winner [59],

TABLE VII
HOW TO DETERMINE THE GSP PAYMENT?

Bidder	Bid	Quality (1-10)	Ranking	Actual Payment
A	B_1:\$2	Q_1:6	B_1*Q_1=12	B_2*Q_2/Q_1=\$1.5
B	B_2:\$3	Q_2:3	B_2*Q_2=9	B_3*Q_3/Q_2
C	B_3	Q_3	B_3*Q_3	...

Given the bid value, some ad networks are able to generate a rough approximation for the number of potential viewers of an ad impression. For example, Facebook allows advertisers to create fine-grained targeted ads and provides an immediate estimation of people who meet the criteria via a real-time "Estimated Reach" box that also includes a detailed report for the ad's future performance [60], [61]. As for quality score, it is predicted using factors such as CTR, ad relevance, landing page quality, load time, and user influence. According to an eye tracking study [45], users commit twice as much visual attention to ads of good quality than to those of poorer quality. For CTRs, good quality ads are estimated to be close to 13% while bad quality ads receive less than 1%. Furthermore, bad quality ads frustrate host publishers [62]. Other factors affecting CTR include influential users. For instance, influential users can persuade other users to contribute to higher CTRs according to AdHeat [63]. And, once clicked, the landing page can then be used to accurately select search ads [64] or automatically recommend bidding keywords [65]. Lastly, most advertisers now use bidding agents during an auction [66], which speeds up the cycle of offline optimization and effectively detects errors in the online bidding phase.

3) Conversion Tracking: Many techniques exist for tracking click-through events that hopefully lead to conversions. For example, in order to track potential conversions such as online

purchases and sign-ups, advertisers can elect to embed analytic code which is provided by an ad network. They may also choose to produce their own code to track click through rates from their landing pages. However, advertisers are more concerned with conversions, which may take place hours or days after the click; yet, research indicates that most conversions occur during the first few hours of a click-through and then quickly drops off afterwards [67]. In cases where advertisers use an ad network's conversion tracking algorithm, the ad network may also utilize a smart pricing algorithm to penalize publishers whose traffic fails to generate conversions [4]. It may even provide discounts to effected advertisers based on a calculated conversion rate, which represents the ratio between impressions achieving conversions and the total number of impressions created.

Other methods use online behavioral advertising (OBA) to track conversions. As its name suggests, OBA is widely used for behavioral targeting, which requires publishers to embed either a tracking pixel or JavaScript code containing the link of a particular ad network. This option enables web tracking by including PII and web content via HTTP across all websites and strengthens collaboration within the ad network. By analyzing historically converted ad campaigns and corresponding metadata, like landing pages, and combining it with user information, Agarwal et al. [68] demonstrate how to automatically and effectively serve targeted users with new or existing ad campaigns to achieve greater conversions. In a comparable work completed by Archak et al. [69], a framework is used to mine the users' long-term behavioral patterns from ads. However, estimates of online advertising are sometimes inaccurate. For instance, some experiments indicate that the importance of observational data is usually overestimated [70]. Similarly, Farahat and Bailey [71] argue that brand interest is the only definitive factor for determining targeted CTRs.

As a result of these capabilities, ad networks are able to individually tailor ads served to users. Indeed, such targeted ads can improve conversion potential for advertisers and potentially increase revenue for the whole ecosystem. For instance, [43] shows that an ad network's revenue may drop by 35-60% when serving unrelated ads to the top 5% of revenue generating users. However, narrowly targeted ads can raise CTRs as much as 670% [60]. According to logs obtained from Bing's search engine [67], advertisers often prefer to correlate bids with a user's prior activity. For instance, they may wish to determine which ad keywords lead to the greatest conversion rates. Thus, when targeting, advertisers consider three factors. With regard to decision making, ad networks either use all or some of these factors. For example, AdMotional [72] integrates all three targeting criteria to personalize their ads while observing and optimizing their rules at varying intervals.

These three factors are:

- advertiser's bid and budget;
- publisher's web content, including contextual information (e.g., web page taxonomies and search queries) and situational information (e.g., user-related spatial and temporal data);
- user's interest and behavioral information (e.g., identity and demographics).

B. Ad Cost & Revenue in Mobile Apps

As of this publication, 73% of apps on the Android marketplace are free [73]. So, why are there so many more free, ad-supported apps, than there are paid, ad-free apps? This question is easily answered if we observe three points. First, the marketplace takes a commission of about 20-30% of the app's price. Second, according to Gartner, users are predicted to increase their downloads of free apps from 90% in 2013 to 93% by 2016 [11]. Likewise, 20% of the free apps receive over 10K downloads, which is in stark contrast to the mere 0.2% of paid apps [73]. Third, from the developer's perspective, app developers may receive continuous income from traditional publishers (with online advertising models) for developing ad-supported apps. Accordingly, the "Legends of Descent" game developers [74] divulge that an ad-supported free app will receive roughly 50 times more downloads and generate roughly 10 times the revenue than its paid, ad-free counterpart. For these reasons, app developers gain higher returns for following free, ad-supported, pricing strategies. Further support for choosing free, ad-supported, apps over paid apps can be found in break-even analysis conducted by another study [75]. They find that the break-even ad income for a free app is around $0.21 per download while a paid, top-selling app is closer to $0.033 per download.

Yet, the story may be even more complex than one might expect. Wei et al. [76] predict that free apps may cost more than their paid counterparts due to their notably higher A&A traffic. Accordingly, Khan et al. [30] evaluates the true cost of a free, ad-supported apps. Considering that the average ad traffic rate in Fruit Ninja is 5.61 Kbps, a Verizon 2GB/$30 plan subscriber will incur a cost of 40MBs of data or 56 cents per month, assuming the game is played every day for 30 minutes. Similarly, Zhang et al. [77] deduce from both free and paid versions of several popular apps in Android and iPhone that more A&A traffic is generated by free apps. Consequently, an AT&T 300MB/$20 plan subscriber using popular free apps (e.g., Dictionary in iPhone or Angry Birds Rio in Android) will spend 48% to 1299% more money than if using a purchased, equivalent app.

Since the top-selling free apps bring even more revenue to their developers with mobile ads, plagiarists repackage these apps and piggyback their own advertising client IDs to steal ad revenue and sometimes implant suspicious payloads [78], [79]. Therefore, AdRob [32] investigates the consequences of such phenomenon with 265K Android apps crawled from 17 app markets along with a 12-day HTTP mobile ad traffic trace from a tier-1, U.S., cellular carrier. After clustering the original apps and their corresponding pirate versions, they show that the original app developers lost an average of 14% in ad revenue and 10% in user population. Of all the apps, Game-related apps suffered the highest percentage of cloning.

Naturally, popular apps and websites affect the whole ecosystem. After monitoring and analyzing four popular third-party Android app marketplaces (i.e., SlideMe, 1Mobile, AppChina, and Anzhi) for over seven months, Petsas et al. [75] find that app downloads perfectly follow the Pareto principle (i.e., the top 10% of apps account for 70% to 90% of total downloads

TABLE VIII
FEATURED NETWORKING-RELATED RESEARCH TOPICS IN DIGITAL ADVERTISING

Subject		Project(s)	Description(s)
Online Advertising	Tracking	[80]	Develop the TrackingTracker add-on to identify trackers over the Alexa top 500 websites, classify web tracking behaviors into five categories more or less related to advertising and offer the ShareMeNot add-on to mitigate web tracking with social widgets by removing third-party cookies
		[81]	Analyze a large-scale dataset from a globally-distributed CDN with the StreamStructure algorithm, which groups streams, detects main objects and identify initial pages, to figure out a growing use of JavaScript for web tracking
	Measuring	[5]	Utilize open recursive DNS platforms to chart three different ad networks (Google's data centers, AOL's CDN and Adblade's standalone server), and PlanetLab to measure network-level (i.e., different delays) and content-level (i.e., ad similarity, location-based and behavioral targeting ads) performances
		[41]	Create a website as the landing page with tracking scripts, subscribe three ad networks (i.e., AdWords, HandyTraffic and Revisitors) to advertise the website in three different countries, and utilize seven metrics to evaluate incoming traffic
		[82]	Collect a large-scale anonymized dataset from Akamai's video delivery network, conduct correlational analysis as well as causal analysis with Quasi Experimental Designs (QEDs) to measure ad completion rate impacted by factors of the ad, of the video and of the viwer, and also study the ad abandonment rate
		[83], [84]	Use a proposed noise-resistant methodology (for measuring OBA by examining differences in Google ads based on web history) to collect ads, detect behavioral targeting in text ads and measure the effectiveness of privacy tools by using cosine similarity with different parameters
Mobile Advertising	Profiling	[31], [38]	Build NetworkProfiler to automatically run Android apps for generating the HTTP traffic and extract fingerprints for identifying the apps, and also identify app usage patterns with the advertising traffic extracted from a Tier 2 cellular service provider
	Tracking	[85]	Recruit 20 participants to use smartphones with the TaintDroid-based AppLog system for three weeks, which dynamically tracks data (including persistent identifiers and web cookies) sent to advertising & analytics over the network

from these four marketplaces). In some cases, the top 1% of apps comprise more than 70% of downloads, which is the case in the Anzhi marketplace. Similar comparisons are found among publishers as well. For instance, among aggregators, the top 5% account for 90% of total revenues [43]. Consequently, a country-level dataset of a cellular network in the above study shows that Facebook receives over 9% of total ad generated revenues. In contrast, Google serves the most users–that being over 18% of total users.

IV. NETWORKING

Within digital advertising, much attention has been given to networking, both for online adverting and mobile advertising. In fact, networking plays an important role in various tracking, measuring, and profiling efforts. Table VIII summarizes the main techniques used for network-related research projects in digital advertising.

A. Tracking & Measuring in Online Advertising

As long as the Internet continues to grow as a popular, complex, and regulated advertising ecosystem, it poses significant challenges for its members. One such challenge for ad networks includes tracking users without violating online privacy policies. In contrast, researchers are challenged to accurately measure how an ad network tracks users since third-party tracking techniques are not publicized by industry. Likewise, security professionals struggle to identify abnormal activities [80].

1) Tracking: Currently, tracking users through a massive population is not technically difficult; however, not everyone is comfortable with being tracked. According to one survey [60], two-thirds of Americans dislike ads that are tailored to their interests, and over 50% prefer to turn off behavioral targeting. In contrast, [81] evaluates the web traffic from 187 countries and shows a growing use of Ajax and JavaScript in an increasing number of objects, which also provide web tracking, within web pages. For example, the use of Google Analytics rose from less than 5% in 2006 to nearly 40% by 2010 with 65% of that dataset's population being tracked. Today, a great number of websites bundle multiple trackers. In response, one group of researchers developed the TrackingTracker add-on [80], which identified over 500 trackers on the Alexa top 500 websites. In one extreme case, over seven trackers were found on a website [80]. Among these trackers are Google Analytics, which tracks within sites, and both DoubleClick and Facebook, which track across sites (i.e., users are tracked when moving from one website to the next). Additionally, a study of 2006 AOL search logs [80] reveals that some trackers can capture over 20% of a user's browsing behavior.

TABLE IX
BROWSER-BASED OPT-OUT SOLUTIONS & DRAWBACKS

Solution	Drawback
Permanent opt-out cookie	Hard to manage hundreds of opt-out cookies
Cookie blocking	Cumbersome to disable cookies site by site
Domain blocking	Difficult to keep updating the blacklist
DNT HTTP header	Just a notification to server, useless [84]
DNT DOM property	Just a notification to browser

In order to opt out of web tracking, regulatory organizations (e.g., FTC) have proposed several browser-based solutions such as permanent opt-out cookie, cookie blocking, domain blocking, Do Not Track (DNT) HTTP header, and Do Not Track DOM property [86], [87]. However, each option has its own drawbacks as seen in Table IX. As a matter of fact, even if these solutions are able to prevent third-party tracking, it is still possible to obtain user profiles from publishers [88]. Yet, third-party tracking without user permission remains an ethical challenge, and recent research indicates that there are many instances where ads are not dependent on user profiles [89]. Thus, the researchers continue to seek balance with tracking technologies

used in web pages, including third-party web services such as ads, analytics, and social plug-ins, with user privacy.

In another survey concerning third-party web tracking [88], non-HTTP cookie implementations are classified as *stateful* (i.e., supercookie) and *stateless* (i.e., fingerprinting) tracking technologies. The supercookie (e.g., local shared object and HTML5 local storage) serves as a persistent alternative to the HTTP cookie. Yet, fingerprinting, a stateless tracking mechanism, can be either active (e.g., social plug-in) or passive (e.g., common identifier). For instance, social plug-ins, such as Facebook's "Like" and Google's "+1", are provided as part of social networking services (SNS) and offered to websites. Often, these services are presented in the form of widgets that can be embedded in a website using an iframe element and maintained in the form of an HTTP cookie. Accordingly, websites can use gathered information to link real identities with social plug-ins.

Likewise, common identifiers such as HTTP referer, content type, and user agent are usually used for grouping HTTP requests [69], [81]. Also, a fingerprint of basic characteristics, such as user agent and time zone, can uniquely identify utilized browsers [90]. Additionally, if a simple combination of metadata (such as gender, birthday, and zip code) are obtained, then studies demonstrate that advertisers can uniquely identify over 60% of the US population [60]. Given 33-bit entropy, this same procedure can uniquely identify a single individual from the world population [60]. Similarly, datasets from Hotmail, Bing, and Windows Update containing cookies and IP addresses are used to determine a host's identity with a precision of 62% if only cookies are used and up to 80% if both are used [91].

2) Measuring: Of those who evaluate online advertising, each does so with a different end state in mind. Advertisers may wish to justify their advertising expenditures in order to validate their costs. For example, a study of a landing page reveals that, regardless of CPC or CPM, the quality of ad traffic is strongly correlated to the price paid by the advertiser [41]. In contrast, Ad networks may want to better understand the ecosystem to enhance business opportunities. For example, the shortest delay for delivering ads occurs when using CDN, and the longest delay occurs when using a standalone ad server [5]. Ad networks may also seek a better approach to extracting advertising keywords from web pages for the purpose of rendering contextual and paid search ads (e.g. URL-based keywords) [92].

Various methodologies exist for measuring online advertising within the research community. In [84], a proposed noise-resistant methodology [83] is used to collect over 80% of Google text ads by reloading a test page seven times, collecting data, and evaluating it with privacy-enhancing tools (like DNT headers and opt-out web pages) across five websites in order to measure their privacy effectiveness. For quality, Zhang et al. [41] utilize seven metrics (i.e., traffic volume, mouse activity, link accesses, user-agent, HTTP referers, timing metrics, and blacklists) to evaluate click traffic. And, while the FourthParty add-on [88] measures dynamic web content, PlanetLab servers are deployed worldwide in order to either measure web traffic [81] and network delay or fetch ad contents [5]. Moreover, Krishnan et al. [82] use large sets of global video delivery data (including 362M videos and 257M ad impressions) from Akamai to measure the effectiveness of *completion rates* and *abandonment rates* of video ads. Furthermore, Iam and Pai [81] observe that up to 12% of all requests (sampled from the top 50 sites for video and advertising) consist of ad related traffic. These percentages tend to increase with market growth.

From one study of three primary ad networks that include Google's large-scale distributed data center network, AOL's CDN-based network, and Adblade's single-server network, researchers found that the lowest global similarity and the highest relevance to location for user browsing patterns occur with ads provided by Google's data centers [5]. Interestingly, [83] finds that keywords from searches still dominate ad delivery, with 73% of ads containing entire search queries and 93% containing at least one word. Likewise, Chen et al. [93] find that revenue increases with behavioral targeting only if the number of comparable advertisers is high enough to create sufficient competition. As for video ads, their completion rates are effected by multiple factors, such as ad position, ad length, video length, and video content [82]. Additionally, their abandonment rates are affected by such factors as the time of viewing and the proportion of the ad watched.

B. Profiling & Tracking in Mobile Advertising

Because of the diversity of user behavior with smartphones [94], almost every service provider tends to collect critical information from handheld devices. Google [76] and Microsoft [12] certainly do so. After collecting and analyzing the network traffic of an Android device using *tcpdump*, Wei et al. [76] reveal that more than 80% of the network traffic for several apps (including Tiny Flashlight, Gasbuddy and Instant Heart Rate) goes to one of the Google services (e.g., maps, ads, analytics, and Google App Engine). Likewise, Vallina-Rodriguez et al. [35] show that Google services in Android account for 73% of ad flow and 80% of total bytes transmitted. Additionally, developers, Obermiller and Bayless [74], suggest collecting as much analytical user information as app developers can in order to improve user retention and increase game revenues. Thus, developers use either their own tracking code or outsource the work to analytics libraries. Gathered statistical data (e.g., user demographics) can also be sent to ad providers upon request in order to conduct mobile user profiling across applications. Ad providers may even use parasitic ad libraries to track users and obtain personal details via Android API calls [27], [29], [33], or their own API calls [40]. In contrast to ad libraries that only help app developers deliver ads, analytics services (e.g., Google Analytics and Flurry [35]) deal with both first and third parties to gather various information from users to improve user experience. The two services, known as A&A services, usually work together to retrieve information that includes the users' location data or app names [95] from mobile traffic.

Mobile tracking uses additional vectors (e.g., cookie, system ID, device ID, SIM card ID, and MAC address) found in A&A traffic [96]. Since many of these are considered to be persistent IDs [29], [85], mobile tracking may also help to discover mobile app usage patterns. For instance, Xu et al. [97] investigate tier-2 cellular network trace and find that weather and news

apps peak in the morning while sports apps peak in the early evening. They also observe that app usage is lowest at noon. As ad traffic is mostly HTTP, unique identifiers assigned by different ad networks can be used not only to identify users, but also to fingerprint apps [38]. For example, Jumptap and Mobclix send Android IDs in clear text so that trackers can easily correlate requests from specific users [29]. Also, NetworkProfiler [31] extracts network fingerprints from Android apps using DroidDriver and Fingerprint Extractor. DroidDriver is able to emulate Android apps and collect their network traces. In cases where there are first-party and third-party traffic, fingerprints can be generated from the hostnames and ad identifiers. Otherwise, DroidDriver can use UI fuzzing to generate events that explore all execution paths. Fingerprint Extractor parses HTTP flows and compares them with the unique and persistent identifiers extracted from other apps. By using NetworkProfiler, researchers successfully identified 306 apps from a dataset obtained over a two-hour block from a cellular network [31].

Unlike online tracking studies, the research community has only just raised concerns about mobile tracking. Grace et al. [27] consider A&A libraries as a whole, but other studies are more specific to analytics. For instance, Applog [85] was implemented from a TaintDroid-like system runtime and an Android app tracking system for sending analytic information to servers. Consequently, it employs 20 participants to obtain tracking statistics. From the statistics, it shows that web cookie, Android ID, and IMEI are the most used in the dataset. While analyzing A&A statistics, the authors also discovered that identifiers are usually sent as plaintext. Beyond mobile app tracking, other studies include mobile web tracking. One example includes the use of a FourthParty-based mobile web privacy measurement tool by Eubank et al [98] to measure dynamic web content against the Alexa top 500 websites on five different mobile devices. These devices are both real and emulated, and include a PC controller. Their experiments show that there are fewer cookies and/or JavaScript calls on mobile devices than on desktop, and only a few mobile specific third-party tracking domains are found. Specifically, the authors discover a growing cookie, which is a third-party cookie that progressively grows in size with repeated encounters while browsing websites, on some mobile devices. Beyond mobile app tracking, location-based services (LBS), such as location sharing, are also found in apps like Foursquare and used to announce a user's location to others. As a result, such services may indirectly advertise for local businesses [99]. This is especially so if users often share the names and locations of restaurants and other establishments they frequent [100].

V. Privacy

In digital advertising, user privacy space is steadily infringed upon. In many cases, users are uncertain as to exactly what privacy they surrender while visiting web pages or using mobile apps. Some companies even specialize in trafficking personal information gleamed from users. This information may be used for marketing or research, but it can also be potentially used for other nefarious purposes. The latter reason has driven researchers to explore new ways to protect user privacy in both online and mobile environments. Table X and Table XI provide an overview of features associated with some current products for online and mobile advertising privacy protection.

TABLE X
Privacy Protection in Online Advertising

Project	Features
ObliviAd [101]	-Hardware-based -Store private data at ad network
[67]	-Software-based -Store private data locally
[102]	-Software-based -Rank ads within the non-tracking environment
Privad [103]–[105]	-Browser add-on -Profile users locally
Adnostic [106]	-Browser add-on -Profile users locally
RePRIV [107]	-Browser add-on -Store user privacy in an encrypted common repository at the publisher side
Priv3 [108], ShareMeNot [80]	-Social add-ons -Remove third-party cookies, but allow the intended elements
SafeButton [109]	-Social add-ons -Store both private and public data locally

TABLE XI
Privacy Protection in Mobile Advertising

Project	Features
AdDroid [112]	-Mobile specific solution -Privilege separation -AdDroid API and two corresponding advertising permissions
AdSplit [39]	-Mobile specific solution -Privilege separation -Two overlapped activities
AFrame [113]	-Mobile specific solution -Privilege separation -Two aligned activities
AppFence [114]	-Mobile specific solution -Two features: data shadowing and exfiltration blocking (based on TaintDroid)
SmartAds [12]	-Mobile specific solution for contextual advertising -Store used keywords with a compact bloom filter -Send the hash values of all keywords
MobiAd [37]	-Traditional solution with an intermediary -Use a common agent to profile users locally -Preserve privacy with a delay-tolerant networking protocol along multiple hops of the path
[115]	-Traditional solution with an intermediary -Use an agent to carry an ad with a match estimator -Make a decision locally with user profile
[116]	-Traditional solution with an intermediary -Use a common agent to group devices and prefetch relevant ads
[73]	-Traditional solution without an intermediary -Send private information separately to developers and ad networks
MoRePriv [117]	-Traditional solution without an intermediary -Use a coarse-grained profile
[118]	-Traditional solution without an intermediary -Select the most relevant ads from a set of ads

A. Online Environment

For individuals with privacy concerns, the Internet is a highly challenging environment. Take for example the cookie churn phenomenon where users receive a cookie, but block it during subsequent visits. Yen et al. [91] reveal that 88% of these cookies are still capable of being tracked. As a result of privacy concerns from various sectors and threats against user privacy, industry continues to advance privacy and security efforts.

According to one study on Google text ads [84], the add-ons (i.e., Ghostery and TACO) and the opt-out tools (i.e., NAI and DAA) are useful for removing or reducing OBA. Additionally, a user can also access and modify their ad preferences through Google Ads Settings. However, these efforts are still insufficient (e.g., Google does not serve ads with HTTPS), and the restrictions are easily bypassed.

Tighter privacy protection is needed for many IT companies as well. This is especially so for well-known IT companies [110], like Google and Facebook. Both perform the dual role of ad network and publisher in order to best monetize their advertising efforts [43], and thus, possess the greatest potential to affect the entire ecosystem. For example, Google sponsored ads are displayed on 80% of all publishers while Facebook's ads are shown on 23% of all publishers (or 85% of the top 10% of publishers) [43]. Their methods for employing ads also vary. For instance, Google AdSense employs MediaBot to crawl web pages for keywords and use them to serve contextually relevant ads [12]. In contrast, Facebook allows advertisers to create fine-grained, targeted ads based on various factors released by users (e.g., age, gender, location, sexual preferences, user demographics and interests), where age and gender are the most important [60], [83]. However, both of these advertising systems may be defective. Castelluccia et al. [111] observe that with the Google Display Network any adversary can infer a Google user's interest categories with just a small number of targeted ads. Their results show that 79% of the inferred categories are correctly reconstructed, and 58% of the original categories are successfully retrieved. Similarly, Korolova [60] explains that with Facebook, any advertiser can infer a user's posted private information with CPM ads or even unposted private information with CPC ads. Although Facebook made efforts with a minimum campaign reach strategy, imposing a minimum target threshold of 20 people, to prevent an attacker from targeting a specific individual, the researcher points out that these efforts are circumvented if an attacker creates 20 Facebook accounts with similar target attributes.

Beyond the above mentioned efforts, the research community has also developed several solutions for user privacy protection. Among these solutions, ObliviAd [101] is a hardware implementation that stores sensitive data on the ad network side, use a secure coprocessor to conduct all ad selections with the oblivious RAM (ORAM) protocol through an encrypted connection and bill advertisers with accumulated tokens. In contrast, Bilenko and Richardson [67] choose an implementation for rendering personalized search ads while retaining a user's behavioral history on the client-side. With this system, newly rendered search ads are matched to recently used keywords in the client-side profile, client-only profile is updated on the server-side with L-BFGS logistic regression, and the probabilities for showing and clicking matched ads become predictable. Also in the non-tracking advertising systems, it becomes harder to isolate the most important factors contributing to CTR. Consequently, expected revenue within the OBA model is often imprecise. For this reason, Reznichenko et al. [102] seek to balance user privacy, when it is applied to an ad ranking, within ad auctions requiring user profiles. Their designs include Rank-at-Client (RaC) and Rank-at-3rd-Party (Ra3), which offer simpler and more efficient methods to counter hard-to-detect cheating by ad brokers. Various add-ons are also being introduced by researchers seeking to combine privacy protection with online advertising. For example, both Privad [103]–[105], and Adnostic [106] choose to profile users locally. With Privad, a dealer is introduced to an ad network and multiple anonymous users and further masks the ad serving and accounting via an encrypted channel so as to protect the anonymity of the users. With Adnostic, the most suitable ad is obtained from a small set of appropriate ads downloaded from the ad network. An impression counter is then used in the CPM model to compute and encrypt statistics in order to prevent the ad network from learning about the user. Another solution, RePRIV [107], permits publishers to mine user interests and behaviors, but stores the private information in an encrypted common repository. This data can then be released to ad networks in a way that aligns with user preferences. Social add-ons or plugins, such as ShareMeNot [80], Priv3 [108], and SafeButton [109], are also being implemented. Both ShareMeNot and Priv3 conditionally remove third-party cookies while loading buttons and allowing selected elements that match the user's intent. As for SafeButton, the social plug-in agent not only maintains its private data locally, which is no more than 150MB for a maximum number of 5K Facebook friends, it also caches publicly accessible data (e.g., the page's total number of "Like" selections). Surprisingly, the render time for presenting combined content with SafeButton is even faster than that of the original Facebook version [109].

B. Mobile Environment

In order to serve more targeted ads and further maximize ad revenue, several parties, including app developers and ad networks, greedily collect user privacy via tracking. For example, the People Hub, a unique feature of Windows Phone that integrates several social networking features, may directly expose user information [117]. Moreover, when ad libraries are bundled within an app and act jointly with a unique and persistent identifier, user privacy is easily leaked and uploaded to remote servers. Furthermore, user privacy is easily compromised due to the lack of secure protocols, like HTTPS, being deployed for fear of the additional overhead created by encryption [29]. Even well-known apps like Facebook for Android permit 22% of its traffic to go unencrypted [76].

Another dataset [96], consisting of over 107K packets of nearly 1.2K apps, uses hierarchical clustering, based on the HTTP packet or destination distances between two packets, to form a dendrogram. Destination distances are calculated using destination IP addresses, port numbers, and host domains along with content distances calculated from the request-line, cookie, and message-body fields of HTTP headers. Using this information, a set of conjunction signatures is generated by clustering the results so as to detect 94% of data leakage while limiting false positives to less than 3%.

Since mobile devices are frequently in a user's possession for calling, messaging, browsing, and other daily activities, maintaining privacy is very important to both users and researchers. The first attempts at revealing the dangers to privacy in mobile advertising originate from a large-scale research investigation of third-party apps. We briefly note a few such studies. By

TABLE XII
COMMON PERMISSIONS

Core permissions used by ad networks	Other dangerous perms used by benign apps	Permissions frequently used by malicious apps
INTERNET	WRITE_EXTERNAL_STORAGE	INTERNET
ACCESS_NETWORK_STATE	RECEIVE_BOOT_COMPLETED	READ_PHONE_STATE
READ_PHONE_STATE	CALL_PHONE	ACCESS_NETWORK_STATE
ACCESS_FINE_LOCATION	CAMERA	WRITE_EXTERNAL_STORAGE
ACCESS_COARSE_LOCATION	READ_CONTACTS	ACCESS_WIFI_STATE
GET_TASKS	GET_ACCOUNTS	READ_SMS
ACCESS_WIFI_STATE	SET_WALLPAPERS	RECEIVE_BOOT_COMPLETED
VIBRATE	CHANGE_WIFI_STATE	WRITE_SMS
READ_LOGS	RECORD_AUDIO	SEND_SMS
WAKE_LOCK	READ_SYNC_SETTINGS	RECEIVE_SMS
	READ_HISTORY_BOOKMARKS	VIBRATE
	READ_CALENDAR	ACCESS_COARSE_LOCATION
	WRITE_CALENDAR	READ_CONTACTS
	WRITE_CONTACTS	ACCESS_FINE_LOCATION
		WAKE_LOCK
		CALL_PHONE
		CHANGE_WIFI_STATE
		WRITE_CONTACTS
		WRITE_APN_SETTINGS
		RESTART_PACKAGES

using 100K Android apps, Grace et al. [27] isolate 100 ad libraries, which cover 52.1% of the entire app set. In another study, Stevens et al. [29] finds 13 popular ad libraries within the top 500 Android apps. Additionally, Book et al. [33] investigate 66 ad library names and versions from 114K free apps in Google Play, and later investigates the top 20 of 103 ad libraries [40]. Findings also indicate that the principle of least privilege is not always followed, since some host apps request additional permissions solely used by ad libraries [119]. In addition, these researchers uncover the behaviors of different ad libraries through various methodologies. For instance, AdRisk [27] statically analyzes the APIs associated with 76 different permissions and refines 14 dangerous APIs used by ad libraries in order to determine the path from an entry point of a dangerous API call to an external sink. In a similar effort [29], Stowaway [16] is used to classify permissions into one of three categories: 1) required, 2) optional and 3) undocumented. Book et al. [33] also contribute to this research using PScout [17] to construct a temporal map depicting changes to permissions used by various ad libraries. Beyond the dangerous API calls in Android and their permissions, Book and Wallach [40] identify 11 ad libraries which use privacy leaking APIs that allow host apps to transmit demographics.

The findings in mobile advertising are more or less related to permissions requested by host apps. Leontiadis et al. [73] highlight a couple of differences between free and paid apps in Android. Case in point, 10% of the free and 40% of the paid apps run without permissions; however, 7% of the free and 1.8% of the paid apps demand more than 10 permissions. Moreover, 73% of the free and 41% of the paid apps request at least one dangerous permission. The free apps dominate in most categories except two (i.e., "Personalization" and "Books & References"). Also, on average, the free apps request 2-3 more permissions than the paid ones in the same category. A study of over 1.8K apps from the SlideMe app store reveals that 73% of the apps have permissions used exclusively by the ad libraries [117]. Moreover, Khan et al. [95] shows that the most used apps (i.e., SNS and IM apps), on which the users spend more than 60% of their time, have no ads. On the other hand, permissions are requested for ad libraries, and the average number of permissions required by an ad network is 3.3 on average [33]. Several research works [29], [33], [39], [73], [117] provide a list of 3-10 core permissions used in ad libraries. These permissions are displayed in Table XII. This list indicates that INTERNET is not the only a mandatory permission and that ACCESS_NETWORK_STATE and READ_PHONE_STATE permissions are widely used in over 70% of ad libraries. The use of these permissions also show a steady increase. Furthermore, other studies [29], [33], [73], [114] identify a few dangerous permissions causing pop-up warnings during installation as shown in Table XII. Coincidentally, most of these dangerous permissions are overlapped with the 12 most abused permissions [120], as shown in Table XIII. In general, these apps follow a bimodal distribution, where both the most and the least popular apps surrender user privacy to ad libraries [40]. As for ad libraries, each app usually embeds more than one ad network–the average being 6.97 ad networks per app [34]. AdMob, a dominant mobile ad library, went as far as to ask developers to explicitly include a user's GPS coordinates [29]. MobClix was another popular ad exchange network that is frequently discussed in research due to its greedy demand for 15 permissions that include the user's contact list, media library, calendar, camera, email, GPS, and SMS [33]. It also requires IMEI and location information [114] and it allows for its advertisers to access other features [27] that include a JavaScript interface to a *WebView* object that exposes additional vulnerabilities [29]. AdMob and MobClix are just two main examples. Other advertising services can be found in similar discussions as well.

Given that both the host app and its ad libraries share the same view with the same permissions, researchers are also exploring countermeasures involving privilege separation in order to internally restrict ad libraries. One proposal is to replace the INTERNET permission with a more fine-grained permission of INTERNET.ADVERTISING (*.admob.com) in order to obtain ads from the AdMob domain [121]. Another

TABLE XIII
12 MOST ABUSED ANDROID APP PERMISSIONS

Permissions	Threats
Network-based Location	Location-based attack or malware
GPS Location	Location-based attack or malware
View Network State	Spot available network connections to download other malware or send text messages, also drain smartphone battery and increase data charges
View Wi-Fi State	Steal Wi-Fi passwords and hack into the networks
Retrieve Running Apps	Steal information from other running apps and "kill" security apps
Full Internet Access	Communicate with command centers, or download updates and additional malware
Read Phone State and Identity	Target device information
Automatically Start at Boot	Autorun malicious apps at every boot
Control Vibrator	Stop vibrations, which can alert users of premium service notifications or verification text messages
Prevent From Sleeping	Prevent phones from going into sleep mode, keep malicious routines running in the background, and drain smartphone battery
Modify/Delete SD Card Contents	Store copies of stolen information before sending to a command center, and delete other personal files
Send SMS Messages	Send messages to premium numbers with unexpected charges, and communicate to command centers

work proposes a privacy control loop to harmonize interests between users and developers by using user-defined permissions in ACCESS_ADVERTISEMENT_SERVICE to separate the host app and ad libraries [73]. Process and/or privilege separation are implemented in three different ways. First, we consider how AdDroid [112] integrates advertising services into the Android system. AdDroid provides a public library API and two corresponding advertising permissions (i.e., ADVERTISING and LOCATION_ADVERTISING) for app developers and uses IPC calls for ad requests. When the system service receives a fetchAd IPC call made by the AdDroid API, it establishes a connection with the proper ad network to get the data, and waits for a follow-up IPC call in order to retrieve the ad. Next we note that AdSplit [39] separates the host app and its ad into two overlapped activities, which allows users to see the ad through a transparent area within the app activity. AdSplit also leverages QUIRE's cryptographic mechanism [18] to detect click fraud. For both apps, a stub library is required to communicate with the ad activity through a standard Android IPC call. Finally, AFrame [113] is an isolated ad displaying activity sharing the same screen with the main activity so that existing ad libraries are still usable without any modification. In addition, AFrame uses an independent graphic buffer to enforce display isolation, and modifies the InputManager to realize the input isolation.

Beyond efforts inside of apps, other solutions attempt to wall off private data from being sent to remote servers. After recognizing that 110 popular and free Android apps were sending location and IMEI to A&A servers, Hornyack et al. [114] implement AppFence, which consists of two primary features for privacy control–those being data shadowing and exfiltration blocking. While the former covertly substitutes sensitive information by sending manipulated data or an empty dataset, the latter, relying on TaintDroid [21], prevents private data from being sent off device by covertly dropping buffered data or overtly simulating an airplane mode state. However, as mobile app users may only pay attention to the ad being rendered when a page is loaded [12], personalized ads are of greater concern to users. This is especially so for impatient users who only spend a few seconds on each app. Therefore, ad targeting is also important alongside privacy preservation and other factors such as overhead.

In fact, contextual advertising in mobile apps is feasible, yet challenging. After characterizing the contents of the top 1200 Windows Phone apps with a UI automation tool, Nath et al. [12] reveal that pages yield more keywords than the metadata (name, category, and description) in over 85% of the mobile apps. Of which, over half the apps dynamically display different contents each time. Thus, in order to enable contextual advertising, it is necessary to extract keywords at runtime. In addition to a framework consisting of a client library and a server, SmartAds [12] improves efficiency and privacy by using a compact bloom filter along with a hash function. The bloom filter stores on the client side about 1MB of the most used keywords and covers 90% of the ads. Additionally, this coverage remains at over 85% after three months. By only sending the hash values of all keywords, the client ensures that the ad server can only learn the keywords fed by the client. Finally, an investigation of 5K ad impressions shows that SmartAds doubles contextually relevant results.

The most used targeting strategy is closely related to user privacy, not app content. However, this strategy does not necessarily sacrifice privacy to improve ad personalization [118]. Inheriting from the heuristics used in online advertising, the solutions for issue trade off can be classified into two categories: with [37], [115], [116] an intermediary or without [73], [117], [118] an intermediary. MobiAd [37] proposes an architecture that serves local ads by using the mobile agent to profile and maintain personal data locally, caching the relevant ads, and preserving privacy with a delay-tolerant networking protocol along multiple hops of the path. Other solutions [115] involve each agent carrying an ad with a match estimator. The match estimator then makes a decision to render an ad by using its access to a locally stored user profile. MASTAds [116] uses an ad management server to group devices with contact graphs into different communities and prefetches sets of relevant ads based on the interests of each community. Another proposal includes market-aware privacy control models allowing users to send private information separately to developers and ad networks in order to balance the flows of private information sent to ad networks for revenue [73]. Furthermore, MoRePriv [117] employs personal preference miners to parse and classify different signals (e.g., Facebook, Twitter, SMS, email, and HTTP traffic) into multiple personas. It then replaces the private information with a coarse-grained profile within apps, using the feature for server-based personalization. Finally, Hardt and Nath [118] develop a framework allowing for mobile devices to select the most relevant ad from a set of ads sent from a server. This selection is based on the estimated CTRs of a large user population.

VI. Security

A. Malvertising in Online Advertising

In 2009, New York Times was attacked by malicious ad-related activity known as malvertising [122]. These activities may be triggered by any party within the ecosystem (i.e., publisher, advertiser, ad network, and even user). This is especially so if ad syndication is used. Ad syndication links partners in the ecosystem by reselling ad impressions from one partner to another in order to eventually form a multi-hop, ad delivery path, which can perpetuate the malvertising problem. Unfortunately, multi-level ad syndication is used by over 75% of malware hosted landing pages [123]. Furthermore, the business model for ad syndication makes malvertising a less-detected occurrence since syndicated ads bypass inspections performed by large and reputable ad networks. After studying the redirection chains of display ads within the Alexa top 90K websites, Li et al. [124] identify the clustering nature of these malicious nodes. As a result of their findings, they implement MadTracer, a topology-based detection system, to detect malvertising activities using existing and newly learned rules. MadTracer's detection rate of real-world malvertising activities proved 15 times more accurate than that of Google Safe Browsing and Microsoft Forefront combined. In addition, they found that over 1% of the sites had already been exploited to provide malicious content [124]. Mavrommatis et al. [123] also state that near 1% of Google's search results are malicious.

Malvertising consists of three types:

- *Scams* use fake anti-virus or lottery phishing to lure people to disclose sensitive information (e.g., usernames, passwords, and bank account numbers).
- *Drive-by Downloads* exploit the vulnerabilities of JavaScript or Flash dynamic contents in browsers or plug-ins.
- *Click Frauds* occur when a script, program, or person masquerades as a legitimate user clicking ads.

A malicious ad network may practice one or more malicious activities; however, both scams and drive-by downloads tend to occur closer to the server by malicious advertisers while click frauds occur closer to the client-side, producing illegitimate clicks and generating revenue for publishers.

Drive-by downloads are the most common malvertising type [125], [126]. From a repository of 66B URLs, Mavrommatis et al. [123] construct a malware distribution infrastructure with over 180K landing pages and 9K related malware distribution sites. They further discovered that third-party ads are a main cause of drive-by downloads. When a user clicks on an seemingly innocuous ad, the clicked window may claim that the user consented to the unintended download and automatic installation. However, since malicious content is commonly hosted on a domain other than the original, protection from such attacks can be accomplished by enforcing a same-origin policy (SOP). The SOP requires web browsers to only load scripts and data from the same origin, which is defined through a combination of protocol type, host name, and port number. Otherwise, access is denied. Also, comparable to drive-by downloads are deceptive downloads. In these attacks, attackers attempts to trick the user into voluntarily downloading unintended software [126].

HTML5/CSS3/JavaScript and Flash are some of the tool sets commonly used in online advertising and drive-by downloading. While AdJail [127] and AdSentry [125] focus on blocking malicious JavaScript ads, OdoSwiff [128] focuses on Flash ads. AdJail forces the SOP to block mutual access between web content and ads. It also relocates ad scripts into a hidden and isolated shadow page and tunnels the mirrored ad content into a real web page in order to display and interact with the ad. AdSentry sandboxes and executes the ad-related JavaScript content within a virtual document object model (DOM) [125]. As for OdoSwiff, static analysis is employed to detect two specific techniques, which are 1) obfuscation and 2) malformed Flash files, used for malicious Flash ads. Additionally, OdoSwiff can employ dynamic analysis to identify anomalous behaviors via an execution trace (i.e., actions and methods, network activity, referenced URLs, and access to the environment) [128].

As for click fraud, it is found in CPC advertising models where ad networks and publishers benefit from illegitimate clicks. For each ad click, money flows from the advertiser directly to the ad network, and ultimately, to the ad publisher. As of 2012, about 10-25% of click frauds were still undetectable [129]. Such operations can be extremely harmful to the ecosystem. As an example, let us consider the FBI's Operation Ghost Click [130]. In spite of netting the largest cybercriminal takedown at that point in history, cybercriminals still managed to extract near to $\$14M$ over a four-year period. Still, click fraud occurs in one of three ways: 1) using clickbots, 2) tricking users into clicking ads (e.g., parked domain monetization), and 3) paying human clickers. All three ways share one common characteristic: click fraudsters receive a higher return on investment (ROI) than do the legitimate publishers. See Table XIV.

There are a number of scientific papers further characterizing the nature of click fraud (e.g., malware [132], C&C [133], DNS hijacking [130], typosquatting [134], Made for AdSense [135], and profile polluter [136]). Usually, the act of click fraud may leave a few clues (e.g., no mouse activities, no subpage visits, no link accesses, and different browser distributions) [41]. However, with emerging botnets, such as TDL-4, Vacha et al. [129] demonstrate how their techniques are able to avoid current detection mechanisms, which include click once per IP address per day, based on user actions, and requirements to use real browsers. Nowadays, click fraud can be initiated by other malware (e.g., Happili [132]). Such malware also includes clickbots, and Miller et al. [133] study two such clickbots (Fiesta and 7cy) in conjunction with a pre-recorded command and control (C&C) dataset and a self-built C&C server that traps outbound clickbot flows. In Fiesta, three CPC components (ad server, search engine, and click server) interact with ad networks. But, 7cy mimics human browsing behaviors at various locations and times. Other research reveals ad fraud activities that change DNS resolution in order to hijack ad impressions and clicks [130]. While ad injectors replace ads provided by legitimate publishers with attacker owned ads in the CPM model, search-hijacking (or link hijacking [126], [137]) techniques are used in the CPC model. Fraud can be initiated by other spoofing techniques as well. After crawling over 285,000

TABLE XIV
COUNTERMEASURES FOR CLICK FRAUD IN ONLINE ADVERTISING

Side		Project	Description
Advertiser		[129]	Use Bayesian estimation as advertisers to measure click-spam rate, and employ graph-clustering over features in the HTTP request as ad networks to detect heavy-hitting clusters
Ad network	Passive	[131]	Test the legitimacy of individual ad clicks actively using either targeted ads with irrelevant display text or the reverse
	Active	[8]	Construct an anomaly detection model with local traffic and features related to cookies and IP addresses, and identify fraudulent publishers when the fraction between the number of suspicious and total requests surpasses a threshold
		ViceROI [4]	Detect fraudulent clicks that falls into an anomalous region outside of an expected log-revenue range and detect six different classes of click-spam attacks

typosquatting domains, Moore and Edelman [134] discovered that 80% of such sites are monetized with CPC ads, which correctly spell the domain names being imitated. According to a study on Made for AdSense (MFA) [135], fraudsters also make use of trending terms to build their MFA websites, so users can be easily lured into their websites through high-ranking search results. Finally, Meng et al. [136] introduce the concept of a profile polluter as an additional fraud mechanism. They demonstrate how publishers, exploiting short term browsing history, can significantly affect re-marketing and behavioral targeting mechanisms for advertising and change the type of ads received by a user. In doing so, the polluter can bias as much as 74% of re-marketed ad impressions and 12% for behavioral ad impressions while yielding up to a 33% increase in revenue for fraudulent publishers.

To defend against the above techniques, several measurement and detection mechanisms have been implemented to minimize click fraud in the ecosystem. For advertisers, Dave et al. [129] utilize a Bayesian framework to independently measure click fraud rates. After conducting a large-scale measurement with ten major ad networks and four types of ads, these researchers discovered that click fraud rates resemble a Bayesian equation based on predefined user actions. For ad networks, countermeasures can be either passive or active. For example, Bluff Ads [131], which are either targeted ads with irrelevant display text or the reverse, can be used to actively test the legitimacy of individual ad clicks. In contrast, Stone-Gross et al. [8] construct an anomaly detection model with features related to cookies and IP addresses. Their model identifies fraudulent publishers if the fraction between the number of suspicious and total requests surpasses a threshold. Furthermore, Dave et al. utilize graph clustering features in HTTP requests to detect "heavy-hitting" clusters, and uncover seven click fraud attacks. Further still, ViceROI [4] is implemented based on the fact that legitimate clicks follow an expected log-revenue range per user while fraudulent ones fall into an anomalous region. Overall, ViceROI, detected six different classes of click fraud.

B. Malfeasance in Mobile Advertising

The diversity of the mobile ecosystem makes it difficult for users to protect themselves on handheld devices. In work conducted by Vidas et al. [138], different attack vectors and countermeasures in Android are surveyed. Similarly, Felt et al. [139] use mobile malware samples to evaluate data detecting techniques and analyze incentives. Moreover, free apps are more intrusive, because 60% of the free apps in Android require Internet connection compared to the 30% required for paid apps [73]. The study of permissions is just one detection technique that we will emphasize below. Following that, we offer a detailed discussion of two incentives: invasive advertising and advertising click fraud.

Normally, original equipment manufacturers (OEMs) do not grant root access rights to mobile users. Therefore, the abilities of each app are constrained by its permissions. As different apps have different functionalities, app developers may request different permissions and mobile users may treat them differently. After developing a knowledge base of mappings between API calls and app behaviors, Rosen et al. [140] used this data set in conjunction with AppProfiler to statically profile apps from mobile users and identify the behaviors of work-oriented apps (e.g., Dropbox and Google Docs). This research determined that work-oriented apps are less susceptible than ad-driven games and SNS apps. In fact, their behaviors are highly correlated with their corresponding permissions. Unfortunately, these permissions can be demanded either by benign or malicious apps, or on behalf of bundled third-party libraries as shown in Table XII. Subsequently, according to other researchers [33], [141], [114], [142], [73], [117], several permissions, such as INTERNET, ACCESS_NETWORK_STATE, WAKE_LOCK, READ_PHONE_STATE, VIBRATE, ACCESS_FINE_STATE, ACCESS_WIFI_STATE, are commonly requested in most cases. Additionally, other permissions, such as GET_ACCOUNTS, RECORD_AUDIO, CAMERA, CHANGE_WIFI_STATE, which are known to leak sensitive information, are also used in ad libraries [33], [114], [140], [73]. However, the behaviors of malicious apps are slightly different from that of benign apps with regard to bundled ad libraries [140]. For example, malicious apps intensively use a few specific permissions (e.g., READ_SMS, RECEIVE_BOOT_COMPLETED, and CHANGE_WIFI_STATE) and greedily demand many more permissions than benign apps, ranging between 4 and 11 [142]. While the strategy of ad libraries is to gather more information, host apps may view them as malware. Therefore, invasive advertising should be avoided. We include the security threats related to in-app mobile ads from [142] in Table XV.

Unfortunately, other inappropriate behaviors of an ad library may result in a supported app being flagged as malware. Here malware is intended to mean a mobile threat with malicious intent while other apps may merely be classified as greyware, which is merely annoying to users. For example, the Apperhand SDK, which collects a wide range of user information and

TABLE XV
SECURITY THREATS FOR IN-APP MOBILE ADS [142]

Behavior	Concept	Example
Repackaging	App reassembled from a popular app with the stealer's advertisingID	AdRob [32]
Drive-by Download	App that entices users to download a malware through an ad link	GGTracker
Remote Control	App that downloads and runs malicious payload to get remote control	Plankton, Kemoge [143]

delivers unwanted notification ads and browser bookmarks, was identified as the "Android.Counterclank" Trojan horse. Still, another ad library, Plankton, which uses DexClassLoader to dynamically and remotely load untrusted Java binary at runtime, is correctly identified as an invasive advertising malware [27], [36], [142], [144]. In order to avoid malware, Android uses permission mechanisms to obtain access authorization from the user. Also, there are a number of mobile anti-virus apps available to users on app store. Meanwhile, the research community continues to make progress towards collecting and detecting malicious samples. For example, Zhou et al. [142] identify approximately 50 malware families from over 1200 malware samples, while DroidRanger [144] uses different schemes to identify both known and unknown malware families. In other efforts, researchers conduct either static (e.g., [141], [145]) or dynamic (e.g., [146], [147]) analysis, or combined (e.g., [148]) analysis in order to study malware. However, cases of malware infection are currently sparse. After studying more than 204K Android apps from five different marketplaces, Zhou et al. [144], observe that infection rates in the official and alternative stores are only 0.02% and below 0.5%, respectively. Also, after analyzing three-months of DNS traffic from a major US carrier, Lever et al. [149] only identify 3,492 infected devices out of over 380M, less than 0.0009% of observed devices.

In malvertising, click fraud on mobile ads is as popular as it is for web ads, depicted in Table XVI. Over 40% of mobile clicks are either accidental (e.g., fat finger) or fraudulent. Thus, industry predictively suffered a loss of nearly $\$1B$ (or 12% of the mobile ad budget) from click fraud in 2013 [150]. In addition, developers may manipulate the screen layout and place ads in regions that cause unintended clicks by real users on embedded ads. That practice, however, definitely violates known policies. For example, in Microsoft Advertising, developers must not "edit, resize, modify, filter, obscure, hide, make transparent, or reorder any advertising". Accordingly, DECAF [150] automatically scans the visual elements of an app and efficiently detects rule violations for Windows-based mobile platforms. Other than placement fraud, developers may employ bots or cheap labor to click on the ads as well. In many cases, mobile applications running in the background are also able to render ads and click on ads without user interaction. These observations led to Crussel et al. developing MAdFraud [151], a system capable of emulating apps, to identify potential fraud. By investigating over 130K apps crawled from the Android market and roughly 35K apps provided by a security company, they find that 30% of these apps completed ad requests while running in the background (possible fraud), 27 of which generated clicks

TABLE XVI
CLICK FRAUD IN MOBILE ADVERTISING

Type	Project
Accidental	Fat finger problem [154]
Fraudulent	DECAF [150], MAdFraud [151], QUIRE [18], AdSplit [39], LayerCake [152], [153]

(likely fraud) [151]. As a result of these findings and the need to protect the advertising ecosystem, user-generated click verification is needed. Fortunately, work in this area has begun. Besides separating the ad activity from the host app to safeguard against a confused deputy attack, QUIRE [18] and AdSplit [39] use an HMAC-based signature in RPC to verify the user-generated click events. Meanwhile, other research provides display and input integrity by implementing a secure user interface that leverages all requests through the system [152], [153]. As a result, implanted ads are protected from being displayed or clicked across trust groups (e.g., the host app).

VII. PEOPLE

As smartphone use becomes a daily habit of people so must advertising on mobile devices take human factors into greater account. Meanwhile, digital ads are no longer limited to a singular person on the Internet. Instead, their influence expands to others connected via social networks. We now discuss this new and growing trend.

A. Human Factors

Since the ultimate goal of mobile ads is to entice customers to click on a presented link, human factors can play an important role. Studies have mostly been human-participation surveys focusing on the effect and weight of various factors. Sometimes, these results may subvert traditional perceptions. There are several such discoveries. For instance, most ad networks do not consider maturity, so mobile ads may display inappropriate content to children [34]. Likewise, clicks may not exactly reflect user intent when users are minors [129]. Additionally, de Sa et al. [155] reveal that blinking animations in mobile ads lowers both user satisfaction and advertising effectiveness. Their work also identifies an inherent relation between user relevance and content relevance. On one hand, content relevance stimulates ad recall, but does little for user experience. On the other, user relevance improves user experience, but has little or no impact on ad recall.

Additional studies focus on user information sharing. Egelman et al. [156] reveals that survey participants are willing to preserve personal privacy by spending more on the apps requiring less permissions, but they will still opt for free, ad-supported apps, at least until required permissions exceed their acceptance threshold. Unlike one traditional assumption in scientific literature, location-related permissions are not considered highly negative [85], [157], [158]. In work completed by Han et al. [85], survey participants indicated that they do not mind being tracked, yet they expect to control the app's settings. Likewise, Kelley et al. [157] find that when allowing access to their location, users believe that their current location

and the quantity of ads are more important than the time of day and advertiser brand. They also find that users are more likely to share their locations on weekdays, while at work, and their second and third most visited places. The first most visited place is assumed to be their home. According to Felt et al. [158], users equate the "sharing of information with advertisers" as a mid-level risk. The high-ranked risks are associated with permanent data or financial loss (e.g., permanently breaking your phone), whereas the low-ranked risks pertain to reversible actions like changing phone settings (e.g., vibrate your phone) or sending data to servers.

B. Social Networks

Most network natives are already familiar with social networking and its prevalence in many aspects of their digital lives. Nowadays, many websites that include videos, online purchases, political commentary, and much more offer users the opportunity to share their purchase, view, or support for political or other causes directly to their social profiles. Findings by Jung et al. also indicate this method of peer influence may have the most significant impacts to attitude and likely behavior across all types of social network advertising (SNA) [159]. Still, social networks are continually looking for new ways to bring their advertisers closer to the right audience, and social media is consistently rife with new changes!

For instance, Facebook's acquisition of the Atlas Ad Platform in 2014 enables cross-platform advertising whether the user is signed into their network or not [160]. This means that digital advertising has now bridged the gap to relevant online impressions for users as they transition from one device to another, be it a smartphone, smartwatch, or network device. This new technology allows Facebook and its advertisers to secure offline purchases as well.

Twitter, another popular SNS, likewise introduced new changes for advertisers allowing them to create ad campaigns with billing that is aligned with their objectives [161]. Last year in 2014 also saw Instagram introduce digital advertising to their platform [161]. As for what is occurring this year in 2015, the following trends are observed [161]:

- *Multi-product ads* which offers advertisers more real estate by depicting three products or three images of the same product in one location.
- *Video ads* showed a 43% increase in 2014 with users watching 38.2 billion videos over one quarter (60% of those videos were watched on smartphones) [162].
- *Local awareness ads* and focus on *Social Local Mobile (SOLOMO)* demonstrates new progress with advertiser brokers seeking to develop groups of customers that meet demographics more in-line with advertiser products.
- *Diversification of social advertising* will bring new players to the social advertising ecosystem, like Instagram, Tumbir, and Pinterest. These new entries may even drive down the overtly promotional posts typically seen on social network sites in favor of more storytelling-type ads.
- *Audience Network* and *Atlas* allow Facebook to move beyond their apps to third-party apps as well while still maintaining targeting features inherent in their platforms.

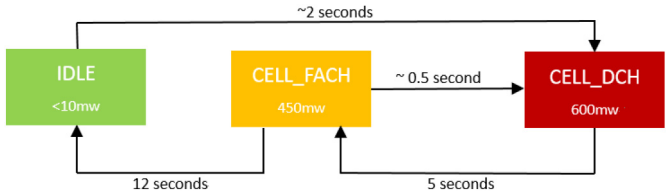


Fig. 4. 3G RRC static machine.

With the addition of Atlas, Facebook is able to extend targeted ads from these apps to websites. Thus, social networking is moving from cookie-based advertising to a cross-platform solution that is integrated with "real-people" across multiple devices.

VIII. Consumption

Handheld devices can get data access from either of two access methods: Cellular Network or Wi-Fi. Accordingly, there are two charging models: by bandwidth and by time. In addition to access and fees, smartphones must also consider power consumption. Khan et al. [95] observe that among 20 users that the average data consumption per user per day is near 30MB, both downloading and uploading. Of which, 44% transits the cellular network. Similarly, Falaki et al. [163] suggest that both cellular network and Wi-Fi should be considered in research to account for both access methods. Therefore, before exploring the consumption of battery energy and cellular data, we will first consider different power modes of each access method.

In 3G cellular networks, radio access network (RAN) dynamically allocates radio resources to the user equipment (UE) via the radio resource control (RRC) protocol. These allocations affect the bandwidth and energy consumption of smartphones. Additionally, even in the same RAN type, different carriers may set different tail times (i.e., delay or inactivity time between two power states). As might be expected, different power states consume power differently. An example of these power states is provided in Figure 4 [35], [164]. Usually, 3G networks defines three power modes: IDLE represents no connections, CELL_DCH (dedicated channel) is in the highest power state with high throughput and low latency, and CELL_FACH (forward access channel) is a transition mode to reduce the mode switching latency.

Whereas 4G LTE has two RRC states, RRC_CONNECTED and RRC_IDLE, as shown in Figure 5. At RRC_CONNECTED state, UE can be in one of the three modes: Continuous Reception, Short Discontinuous Reception (DRX), and Long DRX. While at RRC_IDLE state, UE is only in DRX mode [165].

However, according to [164], Wi-Fi power is far less expensive than cellular power due to its shorter round-trip time (RTT), weaker radio power, and shorter tail time. The Wi-Fi power model, described in [166], uses four states: Deep Sleep (10mW), Light Sleep (120mW), Idle (400mW) and High (600mW). Generally, the Wi-Fi interface regularly receives beacons at intervals of 100ms with power spikes of 250mW from an associated access point. In Deep Sleep, the interface has no communication. But when a packet arrives, the radio

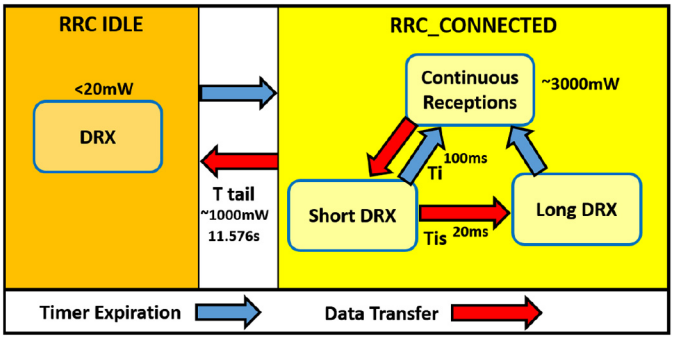


Fig. 5. RRC state transitions in 4G LTE network [165].

immediately moves to the High state. Once the transfer is completed, the radio moves to Idle. The tail time from Idle to Light Sleep is 1s, if no network activity occurs.

Mobile devices still require resource optimization. Unlike traditional feature phones, the batteries on smartphones drain very fast, and overhead traffic further compounds this issue. That waste can be over 200 J/hr with ad traffic accounting for over 87% of total overhead when applications are active and analytics traffic dominating when apps become inactive [77]. According to Pathak et al. [26], only 25-35% of energy is consumed by tested free apps (e.g., Angry Birds and Chess Free). The remaining energy is spent on activities related to advertising, such as ad downloading, user information tracking and updating, and radio tail time. Similarly, research by Mohan et al. [167] finds that mobile ads consume over 65% of communication energy and 23% of total consumption (including CPU, display, and communication). Comparably, Wei et al. [76] observe that there are 50%-100% more system calls occurring in free apps than their paid counterparts, which may result in lower performance and higher battery consumption. For the above reasons, Nicutar et al. [168] focus on optimization techniques such as having nearby users share their links with each other using wireless links (e.g., WiFI or Bluetooth), which have a shorter range and require less power. Kibbutz [168] allows users aggregate their traffic over a subset of links to achieve lower energy consumption as well as shorter RTTs with fairness guarantees for users.

Besides energy consumption, the periodic transfer behavior of ad libraries aggregates data consumption. For example, around 1% of subscribers in a European cellular network spent at least 2 MB/day on ad traffic [35]. This behavior refreshes or re-downloads images, text files and JavaScript code every few seconds. Similarly, Qian et al. [164] find that A&A traffic utilizes about 20% of total cellular data. Typically, the refresh rate of AdMob is between 12 and 120 seconds with a default rate of around 60 seconds [35], [164]. For InMobi, the minimum refresh rate is 20 seconds [35], while both Millennial Media [35] and MobClix [164] refresh every 15 seconds. Since the default tail times of 3G cellular networks, like AT&T, can take up to 17 seconds, these short intervals prevent the smartphone's radio from returning to IDLE [164]. Similarly in 4G LTE as well as Wi-Fi network, the refresh rate makes the smartphone's radio hard to hibernate.

The above issues are further compounded by the following three observations [95]. The first is that the same ads are repeatedly fetched, and the top 100 ads are rendered over 50% of the time. An extremely popular ad might be downloaded up to 82 times on a single device. The second is that about 37% of current ads continue to be served over a 24 hour period and then drop to 14% over the following week. The third is that for a median of 15KB per ad, only a portion (median of 5.8KB) contains ad specific components like ad HTML and images. With regard to the third observation, the remaining portion of the ad consists of redundant JavaScript. Similar redundancy is also found in the actual ad HTML code content. Another study [169] finds that near 20% of the total HTTP traffic is redundant. As a result, their research proposes a prefetching technique to reduce energy and data consumption for mobile ads.

To halve power consumption, AdCache [35], uses prefetching and caching techniques and a 20-second refresh interval on 3G networks for mobile ads, like static banner, animated banner, and text ads. Similarly, CAMEO [30], [95] predictively prefetches mobile ads in bulk while using cheap or free connectivity to support different ad selection models. By observing 1.8M ads consisting of 17K unique ads sent from AdMob across 10 countries, CAMEO shows that it can achieve an accuracy of 84%. Further, CAMEO is able to significantly reduce both energy consumption by 25-37 times and bandwidth by up to 4.8 times when displaying 100 ads.

Yet, due to the constraints of ad campaigns (e.g., budget and duration), prefetching techniques must also ensure SLA compliance and avoid revenue loss. Thus, after evaluating the energy overhead of mobile ads for the top 15 ad-supported Windows Phone apps, Mohan et al. [167] implement mechanisms for app usage prediction and overbooking to model and optimize relationships between prefetching, energy consumption, SLA violations, and revenue loss. After studying users' past behaviors, contained in two datasets, the researchers observe two results. First, for mobile ads with a serving deadline expiring in exactly 30 minutes, the energy consumption can be halved by using a statistical predictor (i.e., the 80th percentile model) along with a prediction interval of 20 minutes while lowering SLA violations to less than 3%. Second, when the deadline is longer, a prediction interval of 15 minutes is enough to satisfy SLAs. Additionally, the background traffic allows for even less energy consumption within this interval. Third, when the deadline is shorter, the overbooking model used in the proxy decreases SLA violations, but increases revenue loss.

IX. Discussion

We now address a few similarities and differences between web-based online advertising and in-app mobile advertising. We also highlight some potential trends and directions for academia and industry in digital advertising.

A. Online Advertising vs. Mobile Advertising

Web-based online ads and in-app mobile ads are two basic forms of digital ads. Yet they possess many distinctions.

According to MobiAd [37], the differences include smaller ads, fewer storage accesses and less frequent downloads for handheld devices in order to compensate for smaller screen sizes, lower processing power and less bandwidth. Indeed, factors such as bandwidth and power usage are less important in online advertising. Therefore, it receives little notice in online advertising studies. But in mobile advertising, these factors are far from negligible. In fact, the bandwidth is determined by the connectivity. While online ads are usually rendered on a web browser with a machine connected to Wi-Fi/Ethernet connection, the mobile ads we study are always displayed within an app on a mobile device connected to Wi-Fi/Cellular connection. As a result, ad traffic flows differently. While ad networks employ client redirection in online advertising, server redirection is used in mobile ecosystems to accommodate the slower connectivity and limited data usage. For this reason, CDNs are also highly relevant in mobile advertising [164]. Since these devices are often in the daily possession of their owners, human factors such as privacy concerns have greater import in mobile environments. While permissions on mobile devices have been studied extensively, little has addressed browser permissions. For example, it might be possible that positioning technologies will improve the effectiveness of online advertising. Whereas in security, the cases of malvertising (e.g., drive-by download and click fraud) frequently appear in online advertising, and there are few cases of ad-related malware in mobile apps [149]. Another distinction is related to accessibility. On one hand, since online ads must comply with the SOP, ad networks have no access to the data on a website. Similarly, a web browser is also completely isolated from other apps on the same machine. And, while third-party web tracking may cover shortages in information collection, it too raises privacy concerns. On the other hand, mobile ads outside of WebView[5] are currently not subject to the privilege separation afforded by web browsers. Since bundled ad libraries share the same set of permissions as host apps, ad networks are able to share user information within the app. Moreover, access to persistent user identifiers (e.g., IMEI, device identifiers, and location information) on mobile devices allows for mobile tracking to extend beyond the boundary of a single app [114], [145]. These approaches allow ad networks to better target their customers; however, contextual advertising, pioneered by Google AdSense, allows ad networks to target customers based on content scraped from the web pages they visit. While crawling web pages is relatively simple, obtaining content at runtime for mobile advertising is still not available. So, new methods are still being pursued. Meanwhile, other technologies are destined to fade. Most smartphones do not support Flash ad technology, so mobile ad developers must continue to embrace new technologies, such as HTML5, if they are to render media rich mobile ads [170]. Table XVII summarizes the differences we witness.

TABLE XVII
DIFFERENCES BETWEEN ONLINE ADVERTISING AND MOBILE ADVERTISING

	Online Advertising	**Mobile Advertising**
Ad Size	Larger	Smaller
Battery	Less Important	More Important
Bandwidth	Faster	Slower
Connectivity	Ethernet/Wi-Fi	Cellular/Wi-Fi
Ad Serving	Client Redirection	Server Redirection
Human Factors	Less Concerned	More Concerned
Permissions	Less Concerned	More Concerned
Infection Rate	Relatively High	Relatively Low
Accessibility	With Browser	Whole Device
Persistence	No	Yes
Flash	Supported	Not Supported
Runtime Crawler	Supported	Not Supported

B. Potential Directions for Industry

Industry is actively and comprehensively working to improve the mobile user advertising experience. Since rich media ads vary in both size and implementation [116], IAB works with the major ad networks to provide a Mobile Rich Media Ad Interface Definition (MRAID) API, which serves to unify many rich media ad formats. Meanwhile, companies like Apple and Google are also working to regulate ad-supported apps internally. For example, the App Store no longer accepts new apps or app updates that access UDIDs as of May 1st, 2013 [171]. In a similar move, Google removed more than 60K apps from Google Play in February 2013 [33]. These low-quality apps usually include ad libraries which demand a high number of permissions [33]. In a similar manner, ad networks are also changing to obfuscate SDKs for intellectual property protection [23]. For example, Android is removing the ACCESS_FINE_LOCATION permission in conjunction with its cancellation of the getLastKnownLocation API [33]. Since users may become annoyed with digital ads, AdBlock Plus and other alternatives are now being widely used. As a result, network advertisers, like Google, have started paying fees to ad blockers to allow their ads [172].

Since solutions such as Cameo [95] and MobiAd [37] require carrier participation, mobile network operators will likely become more heavily involved in the advertising ecosystem in ways that exceed their advertising APIs. As for the underlying operating systems, Android's permissions system still requires substantial changes. Google Play groups app permissions into groups of related permissions during installation that may contain both safe and dangerous permissions [173]. Android's auto-update feature could make the situation even worse, as the existing apps can add new permissions without notifying the user [173]. Such changes may give ad libraries more flexibility and allow advertisers to better reach customers; however, such permissions also come with controversy and potential privacy violations. Finally, it is also possible that Android will adopt the iOS-style permission system for new apps [174]. If so, we envision a progressive update from ad networks.

C. Potential Directions for Academia

We explored multiple points in digital advertising along with many contributions, including the techniques and applications of numerous researchers to enhance the advertising ecosystem. We also note that online ad prefetching techniques are used to

[5]The SOP is still applicable for mobile ads inside WebView [39].

protect user privacy. For example, Privad may be required to cache close to 20MB of ad-related data for daily use [30]. Also, in mobile advertising, prefetching may also be used to optimize energy consumption. Since AdMob gives a grace period of up to 48 hours to obtain reports, greater flexibility is provided to prefetch ads. Another technique used to protect user privacy is privilege separation. However, ad providers and app developers are hesitant to apply it due to the risk of deflating ad revenues. From the beginning of 2015, the research community started looking into ad targeting in mobile advertising [175], [176].

Of equal interest are the new and diverse mobile advertising contributions. For example, AdNext [177] serves highly relevant, location-based ads to users by predicting their next visit based on their behavioral history. Furthermore, AdTouch [178] enables landing page redirection based on QR codes. Lastly, by integrating in-app purchases, in-app mobile ads may benefit users by requiring fewer system modifications.

However, user privacy still suffers from tracking without consent and from unencrypted transmissions. In fact, users may dislike ads due to their passive roles and subsequently push for additional changes. Thus, if users are allowed to make in-app purchases simply by clicking on targeted ads while proactively surrendering private data, all parties (especially advertisers and users) will benefit from such a design. In one case, advertisers will satisfy user demand with sponsored ads; in the other, users will receive financial incentives from viewing the ads. Lastly, the use of a secure channels during in-app purchases will result in protecting user privacy from eavesdropping.

Finally, as a forerunner of mobile advertising, research in online advertising has been conducted for more than a decade. Therefore, the research community has accumulated a substantial set of statistics. A correlation study covering online advertising and mobile advertising may further benefit the domain of digital advertising.

X. Conclusion

Digital advertising is now heavily integrated into many aspects of mobile devices. As a result, people who use smart devices for traditional online browsing and for mobile applications have become an integral part of the digital advertising ecosystem. This ecosystem also consists of brokers, publishers, advertisers, and malicious agents. As such, users have new concerns to consider regarding the use of their personal information (which they trade for services) as well as the power and data constraints of their devices. Additionally, brokers and publishers seek to determine how to best monetize their efforts while meeting advertiser demands without alienating their users. Advertisers also have a vested interest in targeting the right customers for their products to best maximize their advertising investments. These objectives have created unique relationships between all of these parties. Yet, this ecosystem must also remain vigilant in its efforts to thwart rogue agents seeking to compromise these relationships for malicious purposes.

Considering the above concerns, this survey strives to uncover the numerous layers of the advertising ecosystem, ranging from ad cost and revenue generation to privacy and malware concerns. We also focus on in-app mobile advertising and online browsing from mobile devices because these activities represent the growing number of users today. As previously stated, over 60% of video is now watched on mobile devices. Similarly, over 62% of mobile users are using their mobile devices to check email and access the internet while 99.5% access content [179]. Social networking is also driving many changes for how users interact with ads in their mobile app and online viewing.

To further understand the intricacies of digital advertising, this paper explores the background and components of ad networks. Some of which includes advertiser, ad network, publisher, and user relations, CPC objectives, auctions, and privacy protection. Additionally, malvertising and associated protection measures continue to be challenging arenas for researchers. Part of that research includes better security in mobile system functions, the mobile operating systems, and advertising libraries. Additionally, the unique limitations of power and data continue to be explored to reduce unnecessary network traffic, conserve precious mobile power, and reduce unnecessary financial expenditures by mobile users.

Of equal importance is the tracking and profiling limitations of mobile devices, both for targeted advertising and user protection. The fact that mobile devices are often located with their users leads to even greater privacy concerns. Of course, these concerns lead to greater security requirements–policy changes to address them are then added to the advertising ecosystem. To improve security, greater attention to application permissions and library bundles are required. Likewise, better notification for users on what permissions are granted to mobile-app developers is also needed. Again, we highlight the fact that most apps found in Google Play require 3 to 15 permissions. Consequently, additional options for users to opt out of such demands on their privacy are desired.

Since ad networks are also interested in what users are willing to tolerate, the above mentioned issues are of comparable importance to their business models. Multiple surveys discussed in this paper offer suggestions for ad networks and publishers. By paying attention to such literature along with user preferences and developing policies, ad networks can further evolve their business models and tracking techniques to improve marketing, security, and privacy protection. For this reason, it is important that researchers (both in academia and industry) continue to identify new incentives that contribute to the pushes and pulls within the advertising ecosystem. Thus, each advancement towards controlling permissions, protecting privacy, and limiting app malware while gauging and monetizing user interests brings us ever closer to an appropriately balanced advertising ecosystem.

References

[1] L. Goldberg and S. Silber. (Jun. 2014). "At $11.6 billion in q1 2014, internet advertising revenues hit all-time first quarter high." [Online]. Available: http://www.iab.com/at-11-6-billion-in-q1-2014-internet-advertisingrevenues- hit-all-time-first-quarter-high/,

[2] L. Goldberg and S. Silber. (Dec. 2014). "Q3 2014 internet advertising revenues hit $12.4 billion, making it the highest quarter on record." [Online]. Available: http://www.iab.com/q3-2014-internetadvertising-revenues-hit-12-4-billion-making-it-the-highest-quarter-onrecord/,

[3] G. Dunn. (2013). *Cyber-Security and Data Privacy Outlook and Review* [Online]. Available: http://www.gibsondunn.com/publications/pages/Cyber-security-and-Data-Privacy-Outlook-and-Review-2013.aspx

[4] V. Dave, S. Guha, and Y. Zhang, "ViceROI: Catching click-spam in search ad networks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2013, pp. 765–776.

[5] Y. Wang, D. Burgener, A. Kuzmanovic, and G. Maciá-Fernández, "Understanding the network and user-targeting properties of web advertising networks," in *Proc. 31st Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2011, pp. 613–622.

[6] S. Muthukrishnan, "Ad Exchanges: Research Issues," in *Proc. Internet Netw. Econ.*, 2009, pp. 1–12.

[7] R. Gomes and V. Mirrokni, "Optimal revenue-sharing double auctions with applications to ad exchanges," in *Proc. 23rd Int. Conf. World Wide Web*, 2014, pp. 19–28.

[8] B. Stone-Gross, R. Stevens, A. Zarras, R. Kemmerer, C. Kruegel, and G. Vigna, "Understanding fraudulent activities in online ad exchanges," in *Proc. ACM SIGCOMM Conf. Internet Meas. Conf.*, 2011, pp. 279–294.

[9] S. Angel and M. Walfish, "Verifiable auctions for online ad exchanges," in *Proc. ACM Conf. SIGCOMM (SIGCOMM13)*, 2013, pp. 195–206.

[10] A. O. Insider. (2013, Aug.) *How Ad Serving Works* [Online]. Available: http://www.adopsinsider.com/ad-serving/how-ad-serving-works-mobile-vs-web-environments/

[11] C. Pettey and R. van der Meulen. (Jan. 2013). *Global Mobile Ad Market Growing Faster Than Expected, will be Worth us $11.4 Billion in 2013, Double That by 2016, Says Gartner* [Online]. Available: http://www.gartner.com/newsroom/id/2306215.

[12] S. Nath, F. X. Lin, L. Ravindranath, and J. Padhye, "SmartAds: Bringing contextual ads to mobile apps," in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2013, pp. 111–124.

[13] W. Enck *et al.*, "Understanding android security," *IEEE Secur. Privacy*, vol. 7, no. 1, pp. 50–57, Jan./Feb. 2009.

[14] A. P. Felt, H. J. Wang, A. Moshchuk, S. Hanna, and E. Chin, "Permission re-delegation: Attacks and defenses," in *Proc. USENIX Secur. Symp.*, vol. 30, 2011.

[15] A. Shabtai, Y. Fledel, U. Kanonov, Y. Elovici, S. Dolev, and C. Glezer, "Google android: A comprehensive security assessment," *IEEE Secur. Privacy*, vol. 8, no. 2, pp. 35–44, Mar./Apr. 2010.

[16] A. P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner, "Android permissions demystified," in *Proc. 18th ACM Conf. Comput. Commun. Secur.*, 2011, pp. 627–638.

[17] K. W. Y. Au, Y. F. Zhou, Z. Huang, and D. Lie, "PScout: Analyzing the android permission specification," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2012, pp. 217–228.

[18] M. Dietz, S. Shekhar, Y. Pisetsky, A. Shu, and D. S. Wallach, "Quire: Lightweight provenance for smart phone operating systems," in *Proc. USENIX Secur. Symp.*, vol. 31, 2011.

[19] T. Wang, K. Lu, L. Lu, S. Chung, and W. Lee, "Jekyll on iOS: When benign apps become evil," in *Usenix Security*, in presented as part of the 22nd USENIX Security Symposium (USENIX Security 13). Washington, D.C., USA: USENIX, vol. 13, 2013, pp. 559–572.

[20] B. Lau, Y. Jang, and C. Song, "Mactans: Injecting malware into iOS devices via malicious chargers," in *Proc. Black Hat USA*, 2013.

[21] W. Enck *et al.*, "TaintDroid: An information flow tracking system for real-time privacy monitoring on smartphones," *Commun. ACM*, vol. 57, no. 3, pp. 99–106, 2014.

[22] M. Egele, C. Kruegel, E. Kirda, and G. Vigna, "PiOS: Detecting privacy leaks in iOS applications," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, Feb. 2011.

[23] W. Enck, D. Octeau, P. McDaniel, and S. Chaudhuri, "A study of android application security," in *Proc. USENIX Secur. Symp.*, 2011, vol. 2, p. 2.

[24] E. Smith. (2010). *iPhone Applications & Privacy Issues: An Analysis of Application Transmission of iPhone Unique Device Identifiers (UDIDs)* [Online]. Available: URL www.pskl.us/wp/wp-content/uploads/2010/09/iPhone-Applications-Privacy-Issues.pdf

[25] E. Smith. (2012, Sep.) *Apple's UDID Hypocrisy* [Online]. Available: http://www.pskl.us/wp/?p=805

[26] A. Pathak, Y. C. Hu, and M. Zhang, "Where is the energy spent inside my app?: Fine grained energy accounting on smartphones with Eprof," in *Proc. 7th ACM Eur. Conf. Comput. Syst.*, 2012, pp. 29–42.

[27] M. C. Grace, W. Zhou, X. Jiang, and A.-R. Sadeghi, "Unsafe exposure analysis of mobile in-app advertisements," in *Proc. 5th ACM Conf. Secur. Privacy Wireless Mobile Netw.*, 2012, pp. 101–112.

[28] T. Luo, H. Hao, W. Du, Y. Wang, and H. Yin, "Attacks on webview in the android system," in *Proc. 27th Annu. Comput. Secur. Appl. Conf.*, 2011, pp. 343–352.

[29] R. Stevens, C. Gibler, J. Crussell, J. Erickson, and H. Chen, "Investigating user privacy in android ad libraries," in *Proc. Workshop Mobile Secur. Technol. (MoST)*, San Francisco, CA, USA, 2012.

[30] A. J. Khan, V. Subbaraju, A. Misra, and S. Seshan, "Mitigating the true cost of advertisement-supported free mobile applications," in *Proc. 12th Workshop Mobile Comput. Syst. Appl.*, 2012, p. 1.

[31] S. Dai, A. Tongaonkar, X. Wang, A. Nucci, and D. Song, "Networkprofiler: Towards automatic fingerprinting of android apps," in *Proc. INFOCOM*, 2013, pp. 809–817.

[32] C. Gibler, R. Stevens, J. Crussell, H. Chen, H. Zang, and H. Choi, "Adrob: Examining the landscape and impact of android application plagiarism," in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2013, pp. 431–444.

[33] T. Book, A. Pridgen, and D. S. Wallach, "Longitudinal analysis of android ad library permissions," arXiv preprint arXiv:1303.0857, 2013.

[34] Y. Chen, S. Zhu, H. Xu, and Y. Zhou, "Children's exposure to mobile in-app advertising: An analysis of content appropriateness," in *Proc. Int. Conf. Social Comput. (SocialCom)*, 2013, pp. 196–203.

[35] N. Vallina-Rodriguez *et al.*, "Breaking for commercials: Characterizing mobile advertising," in *Proc. ACM Conf. Internet Meas. Conf.*, 2012, pp. 343–356.

[36] V. Svajcer and S. McDonald, "Classifying PUAs in the mobile environment," in *Proc. Virus Bull. Conf.*, VB2013, Berlin, Germany, pp. 139–147, Oct. 2013.

[37] H. Haddadi, P. Hui, and I. Brown, "MobiAd: Private and scalable mobile advertising," in *Proc. 5th ACM Int. Workshop Mobility Evolv. Internet Architect.*, 2010, pp. 33–38.

[38] A. Tongaonkar, S. Dai, A. Nucci, and D. Song, "Understanding mobile app usage patterns using in-app advertisements," in *Passive and Active Measurement*. New York, NY, USA: Springer, 2013, pp. 63–72.

[39] S. Shekhar, M. Dietz, and D. S. Wallach, "Adsplit: Separating smartphone advertising from applications," in *Proc. USENIX Secur. Symp.*, 2012, pp. 553–567.

[40] T. Book and D. S. Wallach, "A case of collusion: A study of the interface between ad libraries and their apps," in *Proc. 3rd ACM Workshop Secur. Privacy Smartphones Mobile Devices*, 2013, pp. 79–86.

[41] Q. Zhang, T. Ristenpart, S. Savage, and G. M. Voelker, "Got traffic?: An evaluation of click traffic providers," in *Proc. Joint WICOW/AIRWeb Workshop Web Qual.*, 2011, pp. 19–26.

[42] O. Chapelle, "Modeling delayed feedback in display advertising," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2014, pp. 1097–1105.

[43] P. Gill, V. Erramilli, A. Chaintreau, B. Krishnamurthy, K. Papagiannaki, and P. Rodriguez, "Best paper–Follow the money: Understanding economics of online aggregation and advertising," in *Proc. Conf. Internet Meas. Conf.*, 2013, pp. 141–148.

[44] H. Beales. (Mar. 2010). "The value of behavioral targeting," Network Advertising Initiative. [Online]. Available: http://www. networkadvertising.org/pdfs/Beales NAI Study.pdf.

[45] G. Buscher, S. T. Dumais, and E. Cutrell, "The good, the bad, and the random: An eye-tracking study of ad quality in web search," in *Proc. 33rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2010, pp. 42–49.

[46] A. Hunter, M. Jacobsen, R. Talens, and T. Winders, "When money moves to digital, where should it go? Identifying the right media-placement strategies for digital display," White paper, comScore and ValueClick Media, Sep. 2010 [Online]. Available: www.recruemedia.com/downloads/comscore2010.pdf, accessed on Jun. 19, 2013).

[47] W. Zhang, S. Yuan, and J. Wang, "Optimal real-time bidding for display advertising," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2014, pp. 1077–1086.

[48] A. Broder, E. Gabrilovich, V. Josifovski, G. Mavromatis, and A. Smola, "Bid generation for advanced match in sponsored search," in *Proc. 4th ACM Int. Conf. Web Search Data Min.*, 2011, pp. 515–524.

[49] H. B. McMahan *et al.*, "Ad click prediction: A view from the trenches," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2013, pp. 1222–1230.

[50] W. Heavlin and A. Radovanovic, "Risk-aware revenue maximization in display advertising," in *Proc. 21st Int. Conf. World Wide Web (WWW'12)*, New York, NY, USA: ACM, 2012, pp. 91–100.

[51] N. Alon, I. Gamzu, and M. Tennenholtz, "Optimizing budget allocation among channels and influencers," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 381–388.

[52] L. C. Stavrogiannis, E. H. Gerding, and M. Polukarov, "Auction mechanisms for demand-side intermediaries in online advertising exchanges," in *Proc. Int. Conf. Auton. Agents Multi-agent Syst.*, 2014, pp. 1037–1044.

[53] B. Lucier, R. Paes Leme, and E. Tardos, "On revenue in the generalized second price auction," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 361–370.

[54] B. Edelman, M. Ostrovsky, and M. Schwarz, "Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords," *Am. Econ. Rev.*, vol. 97, no. 1, pp. 242–259, 2007.

[55] M. Bendersky, E. Gabrilovich, V. Josifovski, and D. Metzler, "The anatomy of an ad: Structured indexing and retrieval for sponsored search," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 101–110.

[56] A. Ghosh and A. Sayedi, "Expressive auctions for externalities in online advertising," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 371–380.

[57] L. E. Celis, G. Lewis, M. M. Mobius, and H. Nazerzadeh, "Buy-it-now or take-a-chance: A simple sequential screening mechanism," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 147–156.

[58] P. Dütting, M. Henzinger, and I. Weber, "An expressive mechanism for auctions on the web," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 127–136.

[59] S. Hua, X. Zhuo, and S.S. Panwar, "A truthful auction based incentive framework for femtocell access," in *Proc. Wirel. Commun. Netw. Conf. (WCNC)*, 2013, pp. 2271–2276.

[60] A. Korolova, "Privacy violations using microtargeted ads: A case study," in *Proc. IEEE Int. Conf. Data Min. Workshops (ICDMW)*, 2010, pp. 474–482.

[61] A. Nath, S. Mukherjee, P. Jain, N. Goyal, and S. Laxman, "Ad impression forecasting for sponsored search," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 943–952.

[62] D. G. Goldstein, R. P. McAfee, and S. Suri, "The cost of annoying ads," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 459–470.

[63] H. Bao and E. Y. Chang, "Adheat: An influence-based diffusion model for propagating hints to match ads," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 71–80.

[64] Y. Choi, M. Fontoura, E. Gabrilovich, V. Josifovski, M. Mediano, and B. Pang, "Using landing pages for sponsored search ad selection," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 251–260.

[65] R. Agrawal, A. Gupta, Y. Prabhu, and M. Varma, "Multi-label learning with millions of labels: Recommending advertiser bid phrases for web pages," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 13–24.

[66] K. J. Lang, B. Moseley, and S. Vassilvitskii, "Handling forecast errors while bidding for display advertising," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 371–380.

[67] M. Bilenko and M. Richardson, "Predictive client-side profiles for personalized advertising," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2011, pp. 413–421.

[68] D. Agarwal, S. Pandey, and V. Josifovski, "Targeting converters for new campaigns through factor models," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 101–110.

[69] N. Archak, V. S. Mirrokni, and S. Muthukrishnan, "Mining advertiser-specific user behavior using adfactors," in *Proc. 19th Int. Conf. World wide web*, 2010, pp. 31–40.

[70] R. A. Lewis, J. M. Rao, and D. H. Reiley, "Here, there, and everywhere: Correlated online behaviors can lead to overestimates of the effects of advertising," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 157–166.

[71] A. Farahat and M. C. Bailey, "How effective is targeted advertising?" in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 111–120.

[72] M. Meyer, M. Balsam, A. O'Keeffe, C. Schlüter, and J. Fendler, "Admotional: Towards personalized online ads," *IJCSA*, vol. 8, no. 2, pp. 59–80, 2011.

[73] 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. 12th Workshop Mobile Comput. Syst. Appl.*, 2012, p. 2.

[74] I. Obermiller and S. Bayless, "Legends of descent: Analytics in an ad-supported windows phone game," in *Proc. 1st ACM Int. Workshop Mobile Gaming*, 2012, pp. 1–6.

[75] T. Petsas, A. Papadogiannakis, M. Polychronakis, E. P. Markatos, and T. Karagiannis, "Rise of the planet of the apps: A systematic study of the mobile app ecosystem," in *Proc. Conf. Internet Meas. Conf.*, 2013, pp. 277–290.

[76] X. Wei, L. Gomez, I. Neamtiu, and M. Faloutsos, "Profiledroid: Multi-layer profiling of android applications," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 137–148.

[77] L. Zhang, D. Gupta, and P. Mohapatra, "How expensive are free smartphone apps?" in *Proc. ACM SIGMOBILE Mobile Comput. Commun. Rev.*, 2012, vol. 16, no. 3, pp. 21–32.

[78] W. Zhou, Y. Zhou, X. Jiang, and P. Ning, "Detecting repackaged smartphone applications in third-party android marketplaces," in *Proc. 2nd ACM Conf. Data Appl. Secur. Privacy*, 2012, pp. 317–326.

[79] W. Zhou, Y. Zhou, M. Grace, X. Jiang, and S. Zou, "Fast, scalable detection of piggybacked mobile applications," in *Proc. 3rd ACM Conf. Data Appl. Secur. Privacy*, 2013, pp. 185–196.

[80] F. Roesner, T. Kohno, and D. Wetherall, "Detecting and defending against third-party tracking on the web," in *Proc. 9th USENIX Conf. Netw. Syst. Des. Implement. (NSDI'12)*, 2012, p. 12 [Online]. Available: http://dl.acm.org/citation.cfm?id=2228298.2228315

[81] S. Ihm and V. S. Pai, "Towards understanding modern web traffic," in *Proc. ACM SIGCOMM Conf. Internet Meas. Conf.*, 2011, pp. 295–312.

[82] S. S. Krishnan and R. K. Sitaraman, "Understanding the effectiveness of video ads: A measurement study," in *Proc. Internet Meas. Conf.*, 2013, pp. 149–162.

[83] S. Guha, B. Cheng, and P. Francis, "Challenges in measuring online advertising systems," in *Proc. 10th ACM SIGCOMM Conf. Internet Meas.*, 2010, pp. 81–87.

[84] R. Balebako, P. Leon, R. Shay, B. Ur, Y. Wang, and L. Cranor, "Measuring the effectiveness of privacy tools for limiting behavioral advertising," in *Proc. Web 2.0 Workshop Secur. Privacy*, 2012.

[85] S. Han, J. Jung, and D. Wetherall, "A study of third-party tracking by mobile apps in the wild," Univ. Washington, Tech. Rep. UW-CSE-12-03-01, 2012.

[86] S. Sprankel, "Online tracking, targeted advertising and user privacy-the technical part," 2011.

[87] M. Bilenko, M. Richardson, and J. Y. Tsai, "Targeted, not tracked: Client-side solutions for privacy-friendly behavioral advertising," in *Proc. 11th Privacy Enhancing Technol. Symp. (PETS11)*, Sep. 2011.

[88] J. R. Mayer and J. C. Mitchell, "Third-party web tracking: Policy and technology," in *Proc. IEEE Symp. Secur. Privacy (SP)*, 2012, pp. 413–427.

[89] P. Barford, I. Canadi, D. Krushevskaja, Q. Ma, and S. Muthukrishnan, "Adscape: Harvesting and analyzing online display ads," in *Proc. 23rd Int. Conf. World Wide Web*, 2014, pp. 597–608.

[90] P. Eckersley, "How unique is your web browser?" in *Privacy Enhancing Technologies*. New York, NY, USA: Springer, 2010, pp. 1–18.

[91] T.-F. Yen, Y. Xie, F. Yu, R. P. Yu, and M. Abadi, "Host fingerprinting and tracking on the web: Privacy and security implications," in *Proc. 19th Annu. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2012.

[92] S. Raju and R. Udupa, "Extracting advertising keywords from url strings," in *Proc. 21st Int. Conf. Companion World Wide Web*, 2012, pp. 587–588.

[93] J. Chen and J. Stallaert, "An economic analysis of online advertising using behavioral targeting," *MIS Quart.*, vol. 38, no. 2, pp. 429–449, 2014.

[94] H. Falaki, R. Mahajan, S. Kandula, D. Lymberopoulos, R. Govindan, and D. Estrin, "Diversity in smartphone usage," in *Proc. 8th Int. Conf. Mobile Syst. Appl. Serv.*, 2010, pp. 179–194.

[95] A. J. Khan, K. Jayarajah, D. Han, A. Misra, R. Balan, and S. Seshan, "CAMEO: A middleware for mobile advertisement delivery," in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2013, pp. 125–138.

[96] H. Kuzuno and S. Tonami, "Signature generation for sensitive information leakage in android applications," in *Proc. IEEE 29th Int. Conf. Data Eng. Workshops (ICDEW)*, 2013, pp. 112–119.

[97] Q. Xu, J. Erman, A. Gerber, Z. Mao, J. Pang, and S. Venkataraman, "Identifying diverse usage behaviors of smartphone apps," in *Proc. ACM SIGCOMM Conf. Internet Meas. Conf.*, 2011, pp. 329–344.

[98] C. Eubank, M. Melara, D. Perez-Botero, and A. Narayanan, "Shining the floodlights on mobile web tracking—A privacy survey," in *Proc. IEEE Workshop Web*, vol. 2, 2013.

[99] I. Polakis, S. Volanis, E. Athanasopoulos, and E. P. Markatos, "The man who was there: Validating check-ins in location-based services," in *Proc. 29th Annu. Comput. Secur. Appl. Conf.*, 2013, pp. 19–28.

[100] J. Lindqvist, J. Cranshaw, J. Wiese, J. Hong, and J. Zimmerman, "I'm the mayor of my house: Examining why people use foursquare-a social-driven location sharing application," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2011, pp. 2409–2418.

[101] M. Backes, A. Kate, M. Maffei, and K. Pecina, "ObliviAd: Provably secure and practical online behavioral advertising," in *Proc. IEEE Symp. Secur. Privacy (SP)*, 2012, pp. 257–271.

[102] A. Reznichenko, S. Guha, and P. Francis, "Auctions in do-not-track compliant internet advertising," in *Proc. 18th ACM Conf. Comput. Commun. Secur.*, 2011, pp. 667–676.

[103] S. Guha, B. Cheng, and P. Francis, "Privad: Practical privacy in online advertising," in *Proc. Symp. Netw. Syst. Des. Implement. (NSDI)*, 2011, pp. 169–182.

[104] S. Guha, A. Reznichenko, K. Tang, H. Haddadi, and P. Francis, "Serving ads from localhost for performance, privacy, and profit," in *Proc. HotNets*, 2009.

[105] H. Haddadi, S. Guha, and P. Francis, "Not all adware is badware: Towards privacy-aware advertising," in *Software Services for e-Business and e-Society*. New York, NY, USA: Springer, 2009, pp. 161–172.

[106] V. Toubiana, A. Narayanan, D. Boneh, H. Nissenbaum, and S. Barocas, "Adnostic: Privacy preserving targeted advertising," in *Proc. Netw. Distrib. Syst. Secur. (NDSS)*, 2010.

[107] M. Fredrikson and B. Livshits, "Repriv: Re-imagining content personalization and in-browser privacy," in *Proc. IEEE Symp. Secur. Privacy (SP)*, 2011, pp. 131–146.

[108] M. Dhawan, C. Kreibich, and N. Weaver, "Priv3: A third party cookie policy," in *Proc. W3C Workshop Do Not Track Beyond*, Oct. 2012.

[109] G. Kontaxis, M. Polychronakis, A. D. Keromytis, and E. P. Markatos, "Privacy-preserving social plugins," in *USENIX Security*, in Presented as part of the 21st USENIX Security Symposium (USENIX Security 12). Bellevue, WA, USA: USENIX, 2012, pp. 631–646.

[110] Fortune. (Feb. 2015). *The World's Most Admired Companies* [Online]. Available: http://fortune.com/worlds-most-admired-companies/apple-1/

[111] C. Castelluccia, M.-A. Kaafar, and M.-D. Tran, "Betrayed by your ads!: Reconstructing user profiles from targeted ads," in *Proc. 12th Int. Conf. Privacy Enhancing Technol. (PETS'12)*, 2012, pp. 1–17 [Online]. Available: http://dx.doi.org/10.1007/978-3-642-31680-7_1

[112] P. Pearce, A. P. Felt, G. Nunez, and D. Wagner, "Addroid: Privilege separation for applications and advertisers in android," in *Proc. 7th ACM Symp. Inf. Comput. Commun. Secur.*, 2012, pp. 71–72.

[113] X. Zhang, A. Ahlawat, and W. Du, "AFrame: Isolating advertisements from mobile applications in android," in *Proc. 29th Annu. Comput. Secur. Appl. Conf.*, 2013, pp. 9–18.

[114] 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. 18th ACM Conf. Comput. Commun. Secur.*, 2011, pp. 639–652.

[115] E. Palme, B. Hess, and J. Sutanto, "Achieving targeted mobile advertisements while respecting privacy," in *Mobile Computing, Applications, and Services*. New York, NY, USA: Springer, 2013, pp. 245–263.

[116] A. Seneviratne, K. Thilakarathna, S. Seneviratne, M. A. Kaafar, and P. Mohapatra, "Reconciling bitter rivals: Towards privacy-aware and bandwidth efficient mobile ads delivery networks," in *Proc. 5th Int. Conf. Commun. Syst. Netw. (COMSNETS)*, 2013, pp. 1–10.

[117] D. Davidson and B. Livshits, "MoRePriv: Mobile OS support for application personalization and privacy," *Microsoft Res.*, vol. 3, May 2012.

[118] M. Hardt and S. Nath, "Privacy-aware personalization for mobile advertising," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2012, pp. 662–673.

[119] X. Wei, L. Gomez, I. Neamtiu, and M. Faloutsos, "Permission evolution in the android ecosystem," in *Proc. 28th Annu. Comput. Secur. Appl. Conf.*, 2012, pp. 31–40.

[120] T. Micro. (2014). *12 Most Abused Android App Permissions* [Online]. Available: http://about-threats.trendmicro.com/us/library/image-gallery/12-most-abused-android-app-permissions

[121] D. Barrera, H. G. Kayacik, P. C. van Oorschot, and A. Somayaji, "A methodology for empirical analysis of permission-based security models and its application to android," in *Proc. 17th ACM Conf. Comput. Commun. Secur.*, 2010, pp. 73–84.

[122] A. Vance. (2009, Sep.) *Times Web Ads Show Security Breach* [Online]. Available: http://www.nytimes.com/2009/09/15/technology/internet/15adco.html?_r=0

[123] N. P. P. Mavrommatis and M. A. R. F. Monrose, "All your iFRAMEs point to us," in *Proc. 17th USENIX Secur. Symp.*, 2008, pp. 1–22.

[124] Z. Li, K. Zhang, Y. Xie, F. Yu, and X. Wang, "Knowing your enemy: Understanding and detecting malicious web advertising," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2012, pp. 674–686.

[125] X. Dong, M. Tran, Z. Liang, and X. Jiang, "AdSentry: Comprehensive and flexible confinement of javascript-based advertisements," in *Proc. 27th Annu. Comput. Secur. Appl. Conf.*, 2011, pp. 297–306.

[126] A. Zarras, A. Kapravelos, G. Stringhini, T. Holz, C. Kruegel, and G. Vigna, "The dark alleys of madison avenue: Understanding malicious advertisements," in *Proc. Conf. Internet Meas. Conf.*, 2014, pp. 373–380.

[127] M. Ter Louw, K. T. Ganesh, and V. Venkatakrishnan, "AdJail: Practical enforcement of confidentiality and integrity policies on web advertisements," in *Proc. USENIX Secur. Symp.*, 2010, pp. 371–388.

[128] S. Ford, M. Cova, C. Kruegel, and G. Vigna, "Analyzing and detecting malicious flash advertisements," in *Proc. Annu. Comput. Secur. Appl. Conf. (ACSAC'09)*, 2009, pp. 363–372.

[129] V. Dave, S. Guha, and Y. Zhang, "Measuring and fingerprinting click-spam in ad networks," in *Proc. ACM Conf. Appl. Technol. Archit. Protoc. Comput. Commun. (SIGCOMM12)*, 2012, pp. 175–186.

[130] S. A. Alrwais, A. Gerber, C. W. Dunn, O. Spatscheck, M. Gupta, and E. Osterweil, "Dissecting ghost clicks: Ad fraud via misdirected human clicks," in *Proc. 28th Annu. Comput. Secur. Appl. Conf.*, 2012, pp. 21–30.

[131] H. Haddadi, "Fighting online click-fraud using bluff ads," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 2, pp. 21–25, 2010.

[132] T. Blizard and N. Livic, "Click-fraud monetizing malware: A survey and case study," in *Proc. 7th Int. Conf. Malicious Unwanted Softw. (MALWARE)*, 2012, pp. 67–72.

[133] B. Miller, P. Pearce, C. Grier, C. Kreibich, and V. Paxson, "What's clicking what? Techniques and innovations of today's clickbots," in *Detection of Intrusions and Malware, and Vulnerability Assessment*. New York, NY, USA: Springer, 2011, pp. 164–183.

[134] T. Moore and B. Edelman, "Measuring the perpetrators and funders of typosquatting," in *Financial Cryptography and Data Security*. New York, NY, USA: Springer, 2010, pp. 175–191.

[135] T. Moore, N. Leontiadis, and N. Christin, "Fashion crimes: Trending-term exploitation on the web," in *Proc. 18th ACM Conf. Comput. Commun. Secur.*, 2011, pp. 455–466.

[136] W. Meng, X. Xing, A. Sheth, U. Weinsberg, and W. Lee, "Your online interests: Pwned! A pollution attack against targeted advertising," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2014, pp. 129–140.

[137] N. Nikiforakis *et al.*, "Stranger danger: Exploring the ecosystem of ad-based url shortening services," in *Proc. 23rd Int. Conf. World Wide Web*, 2014, pp. 51–62.

[138] T. Vidas, D. Votipka, and N. Christin, "All your droid are belong to us: A survey of current android attacks," in *Proc. Usenix Conf. Offensive Technol. (WOOT)*, 2011, pp. 81–90.

[139] A. P. Felt, M. Finifter, E. Chin, S. Hanna, and D. Wagner, "A survey of mobile malware in the wild," in *Proc. 1st ACM Workshop Secur. Privacy Smartphones Mobile Devices*, 2011, pp. 3–14.

[140] S. Rosen, Z. Qian, and Z. M. Mao, "AppProfiler: A flexible method of exposing privacy-related behavior in android applications to end users," in *Proc. 3rd ACM Conf. Data Appl. Secur. Privacy*, 2013, pp. 221–232.

[141] J. Hoffmann, M. Ussath, T. Holz, and M. Spreitzenbarth, "Slicing droids: Program slicing for smali code," in *Proc. 28th Annu. ACM Symp. Appl. Comput.*, 2013, pp. 1844–1851.

[142] Y. Zhou and X. Jiang, "Dissecting android malware: Characterization and evolution," in *Proc. IEEE Symp. Secur. Privacy (SP)*, 2012, pp. 95–109.

[143] Y. Zhang. (2015, Oct.) *Kemoge: Another Mobile Malicious Adware Infecting Over 20 Countries* [Online]. Available: https://www.fireeye.com/blog/threat-research/2015/10/kemoge_another_mobi.html

[144] Y. Zhou, Z. Wang, W. Zhou, and X. Jiang, "Hey, you, get off of my market: Detecting malicious apps in official and alternative android markets," in *Proc. Symp. Netw. Distrib. Syst. Secur. (NDSS)*, Internet Society, 2012.

[145] K. O. Elish, D. D. Yao, B. G. Ryder, and X. Jiang, "A static assurance analysis of android applications," *Virginia Polytechnic Institute and State University*, Tech. Rep., 2013.

[146] V. Rastogi, Y. Chen, and W. Enck, "AppsPlayground: Automatic security analysis of smartphone applications," in *Proc. 3rd ACM Conf. Data Appl. Secur. Privacy*, 2013, pp. 209–220.

[147] Y. Zhang et al., "Vetting undesirable behaviors in android apps with permission use analysis," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2013, pp. 611–622.

[148] Z. Yang, M. Yang, Y. Zhang, G. Gu, P. Ning, and X. S. Wang, "AppIntent: Analyzing sensitive data transmission in android for privacy leakage detection," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2013, pp. 1043–1054.

[149] C. Lever, M. Antonakakis, B. Reaves, P. Traynor, and W. Lee, "The core of the matter: Analyzing malicious traffic in cellular carriers," in *Proc. Symp. Netw. Distrib. Syst. Secur. (NDSS)*, 2013.

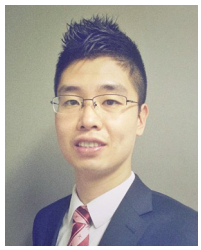
[150] B. Liu, S. Nath, R. Govindan, and J. Liu, "Decaf: Detecting and characterizing ad fraud in mobile apps," in *11th USENIX Symp. Netw. Syst. Des. Implement. (NSDI 14)*, Seattle, WA, USA: USENIX Association, Apr. 2014, pp. 57–70.

[151] J. Crussell, R. Stevens, and H. Chen, "MAdFraud: Investigating ad fraud in android applications," in *Proc. 12th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2014, pp. 123–134.

[152] F. Roesner, J. Fogarty, and T. Kohno, "User interface toolkit mechanisms for securing interface elements," in *Proc. 25th Annu. ACM Symp. User Interface Softw. Technol.*, 2012, pp. 239–250.

[153] F. Roesner and T. Kohno, "Securing embedded user interfaces: Android and beyond," in *Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13)*. Washington, D.C., USA: USENIX, 2013, pp. 97–112.

[154] D. Wigdor, C. Forlines, P. Baudisch, J. Barnwell, and C. Shen, "Lucid touch: A see-through mobile device," in *Proc. 20th Annu. ACM Symp. User Interface Softw. Technol.*, 2007, pp. 269–278.
[155] M. de Sa, V. Navalpakkam, and E. F. Churchill, "Mobile advertising: Evaluating the effects of animation, user and content relevance," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2013, pp. 2487–2496.
[156] S. Egelman, A. P. Felt, and D. Wagner, "Choice architecture and smartphone privacy: There's a price for that," in *The Economics of Information Security and Privacy*. New York, NY, USA: Springer, 2013, pp. 211–236.
[157] P. G. Kelley, M. Benisch, L. F. Cranor, and N. Sadeh, "When are users comfortable sharing locations with advertisers?" in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2011, pp. 2449–2452.
[158] A. P. Felt, S. Egelman, and D. Wagner, "I've got 99 problems, but vibration ain't one: A survey of smartphone users' concerns," in *Proc. 2nd ACM Workshop Secur. Privacy Smartphones Mobile Devices*, 2012, pp. 33–44.
[159] J. Jung, S. W. Shim, H. S. Jin, and H. Khang, "Factors affecting attitudes and behavioural intention towards social networking advertising: A case of facebook users in south korea," *Int. J. Advertis.*, pp. 1–18, 2015.
[160] D. Cohen. (Sep. 2014). *Everything You Need to Know About Facebook's Launch of its Revamped Atlas Ad Platform Social Times* [Online]. Available: http://www.adweek.com/socialtimes/atlas-launch-official/
[161] G. Barbosa. (Jan. 2015). *Top 5 Digital and Social Advertising Trends for 2015* [Online]. Available: https://www.driftrock.com/blog/top-5-digital-social-media-advertising-trends-2015
[162] D. Tweney. (Oct. 2014). Adobe report finds massive 43 growth in online video watching [Online]. Available: http://venturebeat.com/2014/10/20/adobe-report-finds-massive-43-growth-in-online-video-watching/
[163] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin, "A first look at traffic on smartphones," in *Proc. 10th ACM SIGCOMM Conf. Internet Meas.*, 2010, pp. 281–287.
[164] F. Qian *et al.*, "Periodic transfers in mobile applications: Network-wide origin, impact, and optimization," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 51–60.
[165] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck, "A close examination of performance and power characteristics of 4G LTE networks," in *Proc. 10th Int. Conf. Mobile Syst. Appl. services*, 2012, pp. 225–238.
[166] J. Manweiler and R. Roy Choudhury, "Avoiding the rush hours: WiFi energy management via traffic isolation," in *Proc. 9th Int. Conf. Mobile Syst. Appl. Serv. (MobiSys'11)*, 2011, pp. 253–266 [Online]. Available: http://doi.acm.org/10.1145/1999995.2000020.
[167] P. Mohan, S. Nath, and O. Riva, "Prefetching mobile ads: Can advertising systems afford it?" in *Proc. 8th ACM Eur. Conf. Comput. Syst.*, 2013, pp. 267–280.
[168] C. Nicutar, D. Niculescu, and C. Raiciu, "Using cooperation for low power low latency cellular connectivity," in *Proc. 10th ACM Int. Conf. Emerg. Netw. Exp. Technol.*, 2014, pp. 337–348.
[169] F. Qian, "Web caching on smartphones: Ideal vs. reality," in *Proc. 10th Int. Conf. Mobile Syst. Appl. Serv.*, 2012, pp. 127–140.
[170] D. Goldman, "The beginning of the end for adobe's flash," (Nov. 2011). *CNNmoney Tech.* [Online] Available: http://money.cnn.com/2011/11/10/technology/adobe flash/.
[171] M. Panzarino. (2013, Mar.) *Apple to Reject Any Apps That Use UDIDs* [Online]. Available: http://thenextweb.com/apple/2013/03/21/after-a-year-of-warnings-apple-will-no-longer-accept-any-apps-that-//use-udids-as-of-may-1st
[172] L. O'Reilly. (2015, Feb.) *Google, Microsoft, And Amazon Are Paying Adblock Plus Huge Fees to Get Their Ads Unblocked* [Online]. Available: http://www.businessinsider.com/google-microsoft-amazon-taboola-pay-adblock-plus-to-stop-blocking-//their-ads-2015-2
[173] C. Hoffman. (2014, Jun.) *Android's App Permissions Were Just Simplified—Now They're Much Less Secure* [Online]. Available: http://www.howtogeek.com/190863/androids-app-//permissions-were-just-simplified-now-theyre-//much-less-secure/
[174] C. Hoffman. (2015, Oct.) *How to Manage App Permissions on Android 6.0* [Online]. Available: http://www.howtogeek.com/230683/how-to-manage-app-permissions-on-android-6.0/
[175] T. Book and D. S. Wallach, "An empirical study of mobile ad targeting," arXiv preprint arXiv:1502.06577, 2015.
[176] S. Nath, "Madscope: Characterizing mobile in-app targeted ads," in *Proc. 13th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2015, pp. 59–73.
[177] B. Kim *et al.*, "AdNext: A visit-pattern-aware mobile advertising system for urban commercial complexes," in *Proc. 12th Workshop Mobile Computing Syst. Appl.*, 2011, pp. 7–12.
[178] S. Oumtrakul, N. Chanuntawaree, and J. Gao, "Adtouch: A 2D-barcode mobile advertising service system," in *Security-Enriched Urban Computing and Smart Grid*. New York, NY, USA: Springer, 2011, pp. 188–202.
[179] F. N. M. Associates. (Aug. 2012). *A Portrait of Today's Smartphone User* [Online]. Available: https://digitalcontentnext.org/wp-content/uploads/2002/08/MMF-OPA-Portrait-of-Smartphone-User-Aug2012.pdf



Gong Chen (S'15) received the B.Eng. degree in electronic science and technology from Beijing University of Posts and Telecommunications, Beijing, China, in 2007, the Diplôme d'Ingénieur degree in electronics and signal processing from INP-ENSEEIHT, France, in 2010, and the M.S. degree in electrical and computer engineering from Georgia Tech., Atlanta, GA, USA, in 2011. He is currently pursuing the Ph.D. degree in electrical and communication engineering, Georgia Tech, Atlanta, GA, USA. He works with Dr. John Copeland on improving security for digital advertising ecosystems.

Jacob H. Cox, Jr. (S'14) received the B.S. degree in electrical engineering from Clemson University, Clemson, SC, USA, in 2002, and the M.S. degree in electrical and computer engineering from Duke University, Durham, NC, USA, in 2010. He is currently pursuing the Ph.D. degree in electrical and computer engineering at Georgia Institute of Technology, Atalnta, GA, USA. He is an Army Cyber Officer. His research interest include software-defined networking and network security.

A. Selcuk Uluagac (S'01–M'10–SM'12) received the B.S. degree in computer science and engineering from the Turkish Naval Academy, in 1997, the M.Sc. degree in information security from the School of Computer Science, Georgia Institute of Technology, Atlanta, GA, USA, in 2009, the M.Sc. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2002, and the Ph.D. degree in information security and networking from the School of Electrical and Computer Engineering (ECE), Georgia Institute of Technology, in 2010. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Florida International University, Miami, FL, USA. He was a Senior Research Engineer with the School of ECE, Georgia Institute of Technology. Prior to Georgia Tech, he was a Senior Research Engineer with Symantec. He is an expert in securing cyber-physical systems and Internet-of-Things devices. His research interests include cyber security topics with an emphasis on its practical and applied aspects. In 2015, He was the recipient of a Faculty Early Career Development (CAREER) Award from the National Science Foundation (NSF) and was selected to receive Fellowship from the Air Force Office of Sponsored Research (AFOSR)'s Summer Faculty Fellowship Program. In 2007, he was also the recipient of the Outstanding ECE Graduate Teaching Assistant Award from the School of ECE, Georgia Institute of Technology, in 2007.

John A. Copeland (M'67–SM'76–F'83–LF'07) received the B.S., M.S., and Ph.D. degrees in physics from the Georgia Institute of Technology, Atlanta, GA, USA. He holds the John H. Weitnauer, Jr., Chair as a Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology, and is a Georgia Research Alliance Eminent Scholar. He was a VP Techn. with Hayes (1985–1993), and a VP Eng.Techn. with Sangamo Weston, Inc. (1982–1985) and served at Bell Labs (1965–1982). He founded Lancope, Inc. (2000), and invented the StealthWatch network security monitoring system. He has been received 48 patents and has authored more than 100 technical articles. In 1970, he was the recipient of the IEEE's Morris N. Liebmann Award.