Privacy in targeted advertising: A survey

Imdad Ullah [1], Roksana Boreli [2], and Salil S. Kanhere [2]

[1]Prince Sattam bin Abdulaziz University
[2]Affiliation not available

October 30, 2023

Abstract

Targeted advertising has transformed the marketing trend for any business by creating new opportunities for advertisers to reach prospective customers by delivering them personalised ads using an infrastructure of a variety of intermediary entities and technologies. The advertising and analytics companies collect, aggregate, process and trade a rich amount of user's personal data, which has prompted serious privacy concerns among individuals and organisations. This article presents a detailed survey of privacy risks including the information flow between advertising platform and ad/analytics networks, the profiling process, the advertising sources and criteria, the measurement analysis of targeted advertising based on user's interests and profiling context and ads delivery process in both in-app and in-browser targeted ads. We provide detailed discussion of challenges in preserving user privacy that includes privacy threats posed by the advertising and analytics companies, how private information is extracted and exchanged among various advertising entities, privacy threats from third-party tracking, re-identification of private information and associated privacy risks, in addition to, overview data and tracking sharing technologies. Following, we present various techniques for preserving user privacy and a comprehensive analysis of various proposals founded on those techniques and compare them based on the underlying architectures, the privacy mechanisms and the deployment scenarios. Finally we discuss some potential research challenges and open research issues.

Privacy in targeted advertising: A survey

Imdad Ullah, *Member, IEEE,* Roksana Boreli, and Salil S. Kanhere, *Senior Member, IEEE*

Abstract—Targeted advertising has transformed the marketing trend for any business by creating new opportunities for advertisers to reach prospective customers by delivering them personalised ads using an infrastructure of a variety of intermediary entities and technologies. The advertising and analytics companies collect, aggregate, process and trade a rich amount of user's personal data, which has prompted serious privacy concerns among individuals and organisations. This article presents a detailed survey of privacy risks including the information flow between advertising platform and ad/analytics networks, the profiling process, the advertising sources and criteria, the measurement analysis of targeted advertising based on user's interests and profiling context and ads delivery process in both *in-app* and *in-browser* targeted ads. We provide detailed discussion of challenges in preserving user privacy that includes privacy threats posed by the advertising and analytics companies, how private information is extracted and exchanged among various advertising entities, privacy threats from third-party tracking, re-identification of private information and associated privacy risks, in addition to, overview data and tracking sharing technologies. Following, we present various techniques for preserving user privacy and a comprehensive analysis of various proposals founded on those techniques and compare them based on the underlying architectures, the privacy mechanisms and the deployment scenarios. Finally we discuss some potential research challenges and open research issues.

Index Terms—Targeted advertising, Mobile advertising, Online behavioral advertising, Private information retrieval, Privacy, Information leakage, Privacy threats, Tracking, Private advertising systems, Billing, Cryptocurrency, Blockchain, RTB, Characterisation, Obfuscation, Differential privacy.

1 Introduction

The online advertising ecosystem is one of the most successful marketing and advertising markets, operated over with billions of smart devices, including smartphones, tablets, and computer tablets using millions of mobile applications (*apps*) registered in various *app* platforms. The growing need of developing smartphone *apps* has replaced traditional use of internet services; the users are ever more motivated using these *apps* and it is projected that there will be more than 250 billion mobile *apps* downloads by [1]. These mobile *apps* contain at least one *ad library* (including *analytics libraries*) [2] that enables *targeted* (or behavioural) *in-app* mobile advertising. The *advertising and analytics* (A&A) companies use this framework to enable *targeted* advertising, which has also become an increasingly important source of revenue. These companies are competing to increase their revenue by providing *ad libraries* that the *apps* developers use to serve ads to a wide range of audiences.

An important factor in the ad delivery process is the selection of relevant ads to display to relevant users i.e. *targeted* advertising. *Targeted* advertising is based on big data *analytics*, where user's personal information is collected and processed for the purposes of *profiling* and *targeting*. The *in-app* ad control process *targets* mobile users based on various criteria such as device attributes (e.g. OS version, browser type/version etc.), user's temporal behaviour, demographics, interests, *apps*' categories, and location. The assumption in this process is that the *targeted* advertising benefits all parties in an ad ecosystem i.e. the users receiving relevant ads, the *app* developers to receive high ad-refresh/click-through rates, the ad networks to *target* vast majority of audiences, and the advertisers, whose ads are delivered to the appropriate audiences.

Subsequently, the user *profiling*, for effective *targeted* advertising, is becoming increasingly important in the mobile environment, where the above amount of information is collected by mobile *apps* [3] and is sent to *analytics* companies e.g. Google Analytics and Flurry. The *analytics* companies (besides the advertising networks collecting user's personal information) have become an integral part of advertising industry, enabling user *targeting* via their profiles. Such widespread information is collected through the *ad library* API calls [4], including information inference based on monitoring ads displayed during browsing sessions [5], [6]. In the process of *data monetisation*, the ads/analytics companies aggressively look for all the possible ways to gather personal data from the users, including purchasing users' personal data from third parties. This poses serious threats to privacy of users [7], [8], [9], [10], [11], [12], when *apps* indicating sensitive information, e.g., a gaming *app* showing a gambling problem, are the basis for *profiling*.

Therefore, protecting users' personal data with effective *targeting* is important to both the advertising networks and mobile users. The mobile users want to view interest-based ads provided that their information is not shown to the outside world including the adver-

- *I. Ullah is with the College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia. E-mail: i.ullah@psau.edu.sa*
- *R. Boreli was with CSIRO Data61 Sydney, Australia. E-mail: roksana@tmppbiz.com*
- *S. S. Kanhere is with UNSW Sydney, Australia. E-mail: salil.kanhere@unsw.edu.au*

tising companies. This presumption in developing such advertising networks is to deliver most appropriate ads to achieve better view/click through rates and to protect the interactions between mobile users, advertisers and publishers/ad networks.

In this paper, we present a comprehensive survey of existing literature on privacy risks caused due to information flow between the A&A companies, temporal *tracking* of users for their activities and *targeting* them with personalised ads. We describe the *profiling* process, data collection and *tracking* sources for *targeting* users, the ads delivery process, the process of ads characterisation for both *in-app* and *in-browser targeted* ads. Following, we detail the privacy threats posed by the A&A companies as a result of ads *targeting*; in particular, (to prove the privacy leakage) we demonstrate (using experimental evaluation) as how private information is extracted and exchanged among various entities in an advertising system and by third-party *tracking* and the associated privacy risks by *tracking* technologies. Then, we provide various privacy preserving techniques applicable in online advertising e.g. *differential privacy*, *anonymisation*, *proxy-based* solutions, *k-anonymity* i.e. *generalisation* and *suppression*, *obfuscation*, and *crypto-based* techniques such as *private information retrieval* (PIR). Subsequently, we present various privacy preserving advertising systems presented in literature for both *in-app* and *in-browser* using above mentioned techniques and provide their comparative analysis based on the underlying architectures, the privacy techniques used and the deployment scenarios. Finally we discuss some potential research challenges and open research issues.

This article is organised in the following sections: In Section 2, we detail the mobile advertising ecosystem, its operation for ad delivery process, *profiling* process and characterisation of *in-app* and *in-browser* ads. Section 3 presents privacy threats, information leakage in online advertising systems, and various techniques for preserving user privacy from such information leakage are presented in Section 4. Section 5 presents a detailed comparative analysis of various privacy-preserving advertising systems. Various open research issues are outlined in Section 6. Finally, we conclude in Section 7.

2 The Mobile Advertising Ecosystem

The ad network ecosystem involves different entities which comprise of the advertisers, ad agencies and brokers, ad networks delivering ads, *analytics* companies, publishers and the end customers to whom ads are delivered [13]. For the case of large publishers, the ads may be served both by the publishers and the advertisers [14], consequently, the ad ecosystem includes a number of interactions between different parties.

2.1 Information flow between mobile *apps* and ad network

A typical *in-app* mobile ad ecosystem and the information flow among different parties is presented in Figure 1. A user has a number of *apps* installed on their mobile device, that are utilised with specific frequency. As demonstrated in [15], most mobile *apps* include *analytics* Software Development Kit (SDK) and as such both report their activity and send ad requests to the *analytics* and ad network. This network comprises the *Aggregation* server, *analytics* server, *Billing* server, and the *Ads Placement* server. Collected data, that relates to usage of mobile *apps* and the success of displayed ads, is used by the ads *analytics* server to develop user profiles (associated with specific mobile devices and corresponding users). A user profile comprises a number of *interests*, that indicates the use of related *apps*, e.g. sports, business, etc., constructed by e.g., Google **Ad**vertising network for **Mob**ile (AdMob)[1] and Flurry [16] (note that the latter is only visible to *app* developers). *Targeted* ads are served to mobile users according to their individual profiles. We note that other i.e., *generic* ads are also delivered [17]. The *Billing* server includes the functionality related to monetising *Ad impressions* (i.e. ads displayed to the user in specific *apps*) and *Ad clicks* (user action on selected ads).

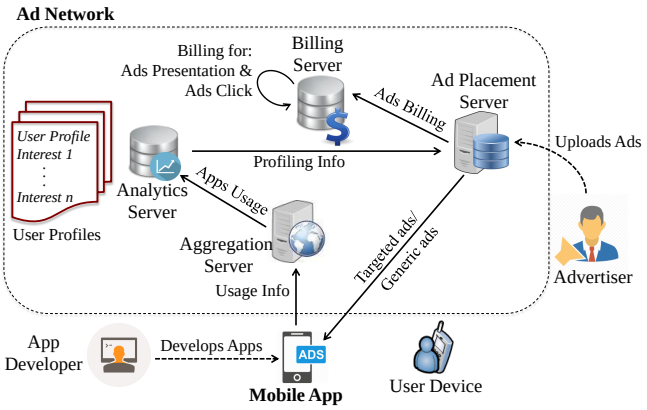

Fig. 1: The *in-app* advertising ecosystem, including the information flow among different parties [17].

2.2 User profiling

Advertising systems rely on user *profiling* and *tracking* to tailor ads to users with specific interests and to increase their advertising revenue. Following, we present the user *profiling* process, in particular, how the user profile is *established*, various criteria, and how it *evolves* over time.

2.2.1 Profile establishment

The advertising companies, e.g., Google, profile users based on the information they add to their Google account, data collected from other advertisers that partner with Google, and its estimation of user's interests based on mobile *apps* and websites that agree to show

1. Google AdMob profile is accessible through the *Google Settings* system *app* on Android devices, accessible through `Google Settings` → `Ads` → `Ads by Google` → `Ads Settings`.

Google ads. An example profile estimated by Google with various demographics (e.g. gender, age-ranks) and profiling interests (e.g. Autos & Vehicles) is shown in Figure 2. It is assumed that there is a *mapping* of the *Apps profile* (the *apps* installed on a user mobile device) to an *Interests profile* (such an example set of interests is shown in Figure 2) defined by advertising (e.g. Google) and *analytics* companies. This information is used by the *analytics* companies to individually characterise user's interests across the advertising ecosystem.

This *mapping* includes the conversion of the *apps* categories Φ_j (where $j = 1, ..., \tau$ and τ is the number of different categories in a marketplace) to interest categories Ψ_l ($l = 1, ..., \epsilon$. ϵ is the number of interest categories defined by the *analytics* company). This *mapping* converts an *app* $a_{i,j} \in S_a$ to interests set $S_g^{i,j}$ after a specific level of activity t_{est}. The t_{est} is the *establishment threshold* i.e. time an *app* should be used in order to establish profile's interests. The result of this *mapping* is a set of interests, called *Interests profile* I_g. Google profile interests[2] are grouped, hierarchically, under vaiours interests categories, with specific interests.

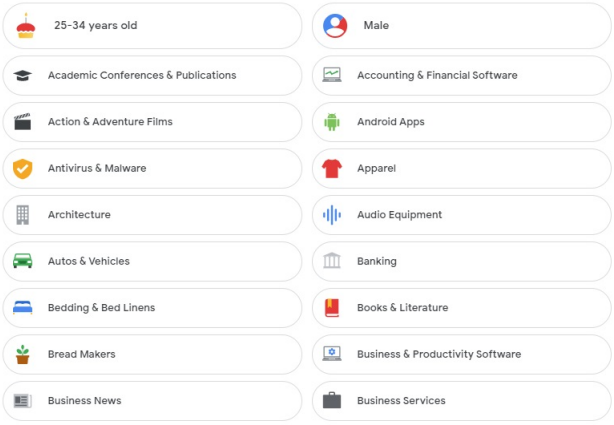

Fig. 2: An (anonymous) example user profile estimated by Google as a results of *Web & App* activity.

In addition, the ads *targeting* is based on demographics so as to reach a specific set of potential customers that are likely to be within a specific age range, gender etc., Google[3] presents a detailed set of various *demographic targeting* options for ads display, search campaigns etc. The demographics D are usually grouped into different categories, with specific options such as age-ranges, e.g. '18-24', '25-34', '35-44', '45-54', '55-64', '65 or more', and gender e.g., 'Male', 'Female', 'Rather not say', and other options e.g. household income, parental status, location etc. The *profiling* is a result of interactions of user device with the AdMob SDK [8] that communicates with Google *analytics* for deriving user profiles. A complete set of '*Web & App* activities' can be found under 'My Google Activity'[4], which helps Google make services more useful, such as, helping rediscover the things already searched for, read, and watched.

Figure 3 shows, a specific example of Google, various sources/platforms that Google use to collect data and *target* users with personalised ads. These include a wide range of different sources enabled with various tools, e.g., the '*Web & Apps* activities' are extracted with the help of Andoird/iOS SDKs, their interactions with *analytics* servers within Google network, cookies, *conversion tracking*[5], web searches, user's interactions with received ads etc. Google Takeout[6] can be used to export a copy of contents (up to several GBs of data) in user's Google Account for backup or use it with a service outside of Google. Furthermore, this includes the data from a range Google products personalised for specific users that a user use, such as, email conversations (including *Spam* and *Trash* mails), contacts, calendar, browsing & location history, and photos.

2.2.2 Profile evolution

The profile is updated, and hence the ads *targeting*, each time variations in the users' behaviour are observed; such as for a mobile user using *apps* that would map to interests other than the existing set of interests. Let a user uses a new set of *apps* S'_a, which has no overlap with the existing set of *apps* S_a that has created I_g i.e., $S'_a \subset \mathcal{A} \setminus S_a$, $\mathcal{A}$ is the set of *apps* in an *app* market. The newly added set of *apps* S'_a is converted to interests with t_{evo} as *evolution threshold* i.e. the time required to evolve profile's interests. Hence, the final *Interests profile*, I_g^f, after the *profile evolution* process, is the combination of older interests derived during the profile *establishment* I_g and during when the profile *evolves* I'_g.

2.2.3 Profile development process

In order for the *Apps profile* to *establish* an *Interests profile*, a minimum level of activity of the installed *apps* is required. Furthermore, in order to generate one or more interests, an *app* needs to have the AdMob SDK. We verified this by testing a total of 1200 *apps* selected from a subset of 12 categories, for a duration of 8 days, among which 1143 *apps* resulted the *Interest profiles* on all test phones indicating "Unknown" interests. We also note that the *Apps profile* deterministically derives an *Interests profile* i.e., a specific *app* constantly derives identical set of interests after certain level of activity. We further note that the level of activity of installed *apps* be within a minimum of 24hours period (using our extensive experimentations; we note that this much time is required by Google *analytics* in order to determine ones' interests), with a minimum of, from our experimentations, $24/n$

2. Google profile interests are listed in https://adssettings.google.com/authenticated?hl=en, displayed under the 'How your ads are personalized'. Note that Google services can also be verified on Google Dashboard https://myaccount.google.com/dashboard?hl=en.

3. Demographic Targeting https://support.google.com/google-ads/answer/2580383?hl=en

4. https://myactivity.google.com/myactivity?otzr=1

5. https://support.google.com/google-ads/answer/6308

6. https://takeout.google.com/

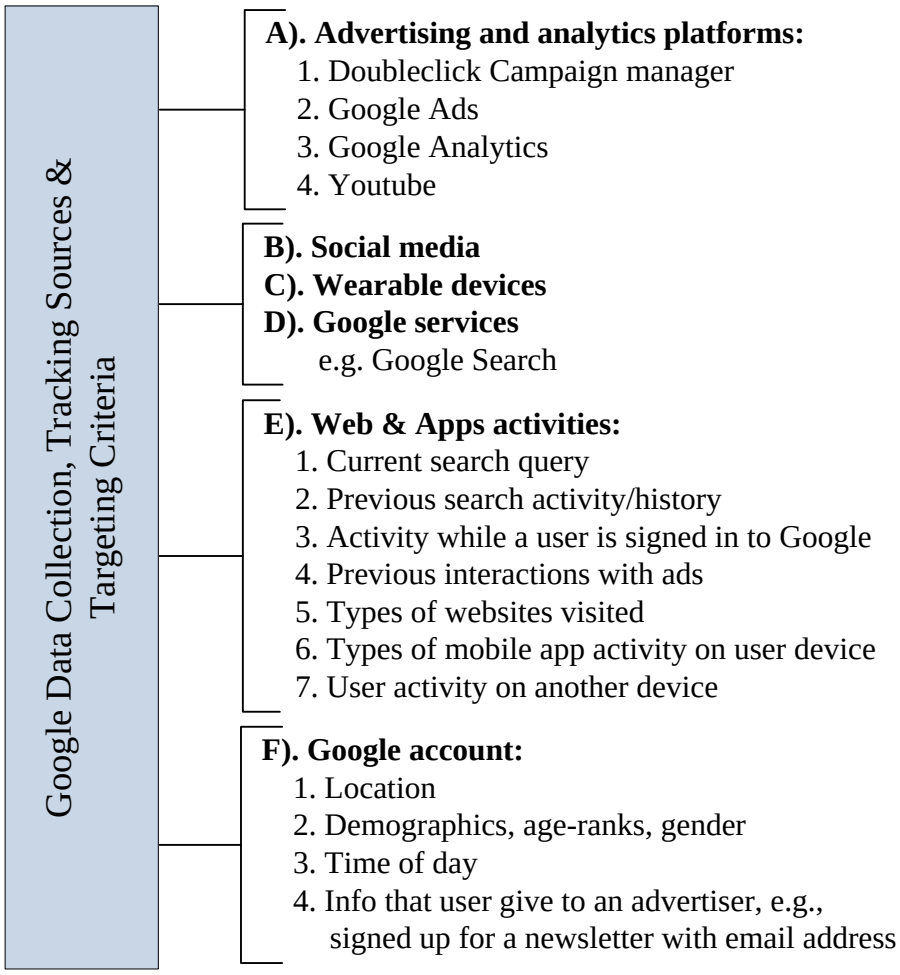

Fig. 3: Google's data collection and *tracking* sources for *targeting* users with personalised ads.

hours of activity of n *apps*. For a sophisticated *profiling*, a user might want to install and use a good number of *apps* that would represent one's interests. After the 24hours period, the profile becomes *stable* and further activity of the same *apps* does not result in any further changes. The mapping of *Apps profile* to *Interests profile* during the *establishment* and during the *evolution* process along with their corresponding *stable* states are shown in Figure 4.

Similarly, during the profile *evolution* process, the *Interests profile* starts changing by adding new interests; once *apps* other than the existing set of *apps* S_a are utilised. However, instead of 24hours of period of evolving a profile, we observe that the *evolution* process adds additional interests in the following 72hours of period, after which the aggregated profile i.e. I_g^f becomes *Stable*. In order to verify the stability of the aggregated profile, we run these *apps* on 4th day; henceforth we observe no further changes. The mapping of *Apps profile* to *Interests profile* during the *establishment* and during the *evolution* process along with their corresponding *Stable* states are shown in Figure 4.

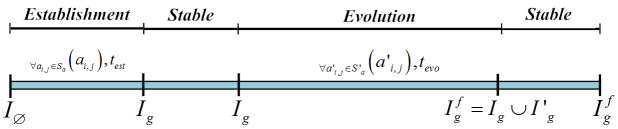

Fig. 4: Profile *establishment* & *evolution* processes. $I_\emptyset$ is the empty profile before *apps* utilisation. During the *stable* states, the *Interest profiles* I_g or I_g^f remains the same and further activities of the same *apps* have no effect over the user profiles.

2.3 Ad delivery process

We identify the workflow of a mobile *app* requesting a Google AdMob ad and the triggered actions resulting from e.g. a user click (we note that other advertising networks, such as Flurry, use different approaches/messages to request ads and to report ad clicks). Figure 5 describes some of the domains used by AdMob (Google ad servers and AdMob are shown separately for clarity, although both are acquired by Google). As shown, an ad is downloaded after the `POST` method is sent by mobile phone (Step 2) containing

phone version, model, *app* running on phone etc. The ad contains the landing page (web address of an ad-URL) and `JavaScript` code that is executed where some of the static objects are downloaded (such as a `PNG`, (Step 3)). Two actions are performed after clicking an ad: a *Conversion* cookie[7] is set inside phone (Step 4) and the web server associated with the ad is contacted. The landing page may contain other list of servers (mainly residing in Content Delivery Networks) where some of the static objects are downloaded and a complete `HTML` page is shown to the user (Step 5). The mobile *apps* developers agree on integrating ads in mobile *apps* and the ads are served according to various rules set by the ad networks, such as to fill up their advertising space, and/or obtaining *profiling* information for *targeting*. Additionally, the ads refreshment intervals, mechanisms used to deliver ads (push/pull techniques), the strategy adopted after ad is being clicked, and click-through rates etc. are also defined by the ad networks.

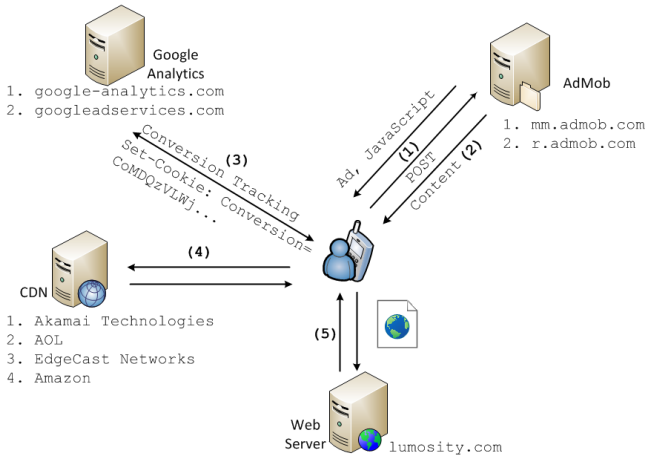

Fig. 5: AdMob Ad Presentation Workflow [17].

In consequence, the ad networks are complex systems being highly diverse with several participants and adopting various mechanisms to deliver ads. Thus, in order to correctly identify and categorise ads and to server appropriate ads, it needs to investigate various ad delivery mechanisms and also cope with such diversity. This evaluation process needs identifying and collecting various ads delivery mechanisms through inspecting collected traffic traces captured from several *apps* executions, as shown in Figure 5. In addition, it needs to emphasis on ads distribution mechanisms used by ad networks from the *apps*' perspective or users' interests to devise the behaviour of ads pool served from ad networks and how they map to individual user's interest profiles. Since the advertising system is a closed system, this process needs to indirectly evaluate the influence of different factors on ad delivery mechanisms, which is even more complicated in Real Time Bidding (RTB) scenarios and associated privacy risks.

7. *Conversion tracking* is specifically used by Google that is an action a customer takes on website that has value to the business, such as a purchase, a sign-up, or a view of a key page [18].

2.4 Understanding ad network's operation

The advertising networks provide an SDK for integrating ads inside the mobile *apps* while securing the low level implementation details. The ad networks provide regulation for embedding ads into the mobile *apps*, the ad delivery mechanism, the amount of times an ad is displayed on the user screen and how often an ad is presented to the user. The common type of ad is the flyer, which is shown to the user either at the top or at the bottom of device's screen, or sometimes the entire screen is captured for the whole duration of the ad presentation. These flyers are composed of text, images and the `JavaScript` codes.

The ad presentation workflow of Google AdMob is shown in Figure 1 that shows the flow of information for an ad request by an *app* to AdMob along with the action triggered after the user clicks that particular ad. This figure shows the `HTTP` requests and the servers (i.e. Content Delivery Network (CDN) or ad servers) used by AdMob. Furthermore, several entities/services and a number of `HTTP` requests to interact with the ad servers and user agent can be observed in this figure.

2.5 Ads selection algorithms

The accurate measurement of the *targeted* advertising is systematically related to the *ad selection* algorithm and is highly sensitive since it combines several fields of mathematics, statistics, analytics, and optimisation etc. Some of the *ad selection* algorithms show *ad selection* based on the user data pattern [19] and the program event analysis [20], however, the *contextual* and *targeted* advertising is treated in different way as they are related to the psyche of the users. Consequently, it has been observed that the activity of users and their demographics highly influences the *ad selection* along with the user clicks around an ad [21], [22]. As an example, a young female that is frequently browsing websites or using mobile *apps* related to the category of *entertainment*, would be more interested in receiving ads related to *entertainment* such as movies, musical instruments etc., consequently, it increases the *click-through rates*. Another work [23] builds a game-theoretic model for ad systems competing through *targeted* advertising and shows how it effects the consumers' search behavior and purchasing decisions when there are multiple firms in the market. We note that the researchers utilise different *ad selection* and *targeting* algorithms based on machine learning and data mining techniques.

2.6 Ad traffic analysis

2.6.1 Extracting ad traffic

Recall that the mobile ad network involves different entities to interact during the ad presentation and after

an ad is being clicked to download the actual contents of the ad, as observed in Figures 1 and 5. Specifically, these entities are the products, the ad agencies attempting ad campaigns for the products, ad networks delivering ads, the publishers developing and publishing mobile *apps*, and the end customer to whom ads are delivered [13]. It is likely, when it comes to large publishers, that both the publishers and advertisers may have their own ad servers, in which case, some publishers may configure to put certain ads pool on the advertisers' side and, at the same time, maintain their own ad servers [14]. The publishers, this way, can increase their revenue by means of providing redundant ad sources as if one ad network fails to deliver ads then they can try another ad network to continue providing services. In similar way, an end user may experience to be passed over several ad networks from publishers to the advertisers to access ads.

2.6.2 Ads traffic identification

The advertising system itself and its functionality are very diverse and complex to understand its operation [7], hence in order to categorise the ad traffic, it needs to be able to incorporate such diversity. This can be performed by first capturing the traces from the *apps* that execute and download the ad traffic and then investigating the traffic characteristics. Characterising and inspecting the ad traffic can give information about the approaches used by multiple publishers, the various mechanisms used to deliver ads by the publishers, the use of different ad servers, and the ad networks themselves [24]. Similarly, it helps identify any *analytics* traffic used by the ad networks to *target* with relevant ads. Analysis of the traffic traces enables to parse and classify them as traffic related to **i)** ad networks, **ii)** the actual web traffic related to ad, **iii)** traffic related to CDN, **iv)** *analytics* traffic, **v)** tracking traffic, **vi)** ad auctions in RTB, and **viii** statistical information about apps usage or developer's statistics, and **ix)** traffic exchange during and after an ad click. As a consequence, a major challenge is to be able to derive comprehensive set of mechanisms to study the behaviours of ad delivery, classify the connection flows related to different ad networks, detecting any other possible traffic, and to classify them in various categories of ads.

2.6.3 In-app mobile vs. in-browser web ads traffic analysis

We note that there are several differences in separately collecting and analysing the *in-app* and *in-browser* user's advertising ad/data traffic for the ad delivery mechanism in order to *target* users. Analysing the *in-app* ad traffic requires to be able to derive comprehensive set of rules to study the ad delivery behaviours (since several ad networks adopt their own formats for serving ads, as mentioned above), catalogue connection flows, and to classify ads categorisation. Furthermore, the ad delivery mechanisms are not publicly available, hence, analysing *in-app* ads would be dealing with an inadequate information problem. Although *in-browser* ad delivery mechanism can be customised[8] to receive ads which are tailored to a specific profiling interests [25], [26].

For the *in-app* ads delivery [7], [8], [27], [28], [29], an ad network may use different information to infer users' interests, in particular, the installed applications together with the device identifier to profile users and to personalise ads pool to be delivered. Similarly, for *in-browser* ads, user *profiling* is performed by *analytics* companies [30] through different information such as browsing history, web searches etc., that is carried out using configured cookies and consequently *target* users with personalised ads. However, in *in-app* ads context, this information might be missing, or altogether not permitted by the OS, as the notion of user permissions may easily prevent the access to data out of the *apps* environment.

2.6.4 Targeted advertising

The *in-app targeted* advertising is a crucial factor in increasing revenue (a prediction shows the mobile ad market to grow to $408.58 billion in 2026 [31]) in a mobile *app* ecosystem that provides free services to the smartphone users. This is mainly due to users spend significantly more time on mobile *apps* than on the traditional web. Hence, it is important (note that *targeted* advertising is not only unique to the *in-app* but has also been used for *in-browser* to deliver ads based on user's interests. The characterisation of *targeted* advertising, on the user's side, is the in-depth analysis of the ad-delivery process so as to determine what information the mobile *apps* send to the ad network and how effectively they utilise this information for ads *targeting*. Furthermore, the characterisation of *in-app* mobile ads would expose the ad-delivery process and the ad networks can use the resultant analysis to enhance/redesign the ad delivery process, which helps in better view/click through rates.

It is crucial for the *targeted* advertising to understand as what information do *apps* (both free and paid mobile *apps* of various categories) send to the ad networks, in particular, how effectively this information is used to *target* users with interest-based ads? whether the ad networks differentiate among different types of users using *apps* from the same or different *apps* categories (i.e. according to *Apps profile*)? how much the ad networks differentiate mobile users with different profiles (i.e. according to *Interests profile*)? the effect over user *profiling* with the passage of time and with the use of *apps* from diverse *apps* categories (i.e. during profile *evolution* process)? the distribution of ads among users with different profiles? and the frequency of unique ads along with their ads serving distributions?

8. E.g. by modifying Google ads preferences: https://adssettings.google.com/authenticated?hl=en

2.7 Characterisation of *in-app* advertisements

There is a limited research available on characterising the *in-app* ads. Prior research works have demonstrated the large extent to which *apps* are collecting user's personal information [13], the potential implications of receiving ads to user's privacy [6] and the increased utilisation of mobile device resources [14], [32]. In our previous study [17] (and in [33]), we observe that various information sent to the ad networks and the level of ads *targeting* are based on communicated information, similarly, we [9] investigate the installed *apps* for leaking targeted user data. To combat these issues, a number of privacy preserving [25], [26], [34] and resource efficient mobile advertising systems [14], [32] have been proposed. Works on the characterisation of mobile ads have primarily focused on measuring the efficiency of *targeted* advertising [21], to examine whether the *targeted* advertising based on the users' behaviour leads to improvements in the *click-through rates*. However, thus far there have been limited insights about the extent to which *targeting* is effective in mobile advertising that will ultimately determine the magnitude of various issues such as bandwidth usage, including the loss of privacy.

We note that existing approaches on characterising *targeted* advertisements for *in-browser* [6], [21], [25], [26], [35], [36], [37], [38], [39], [40] cannot be directly applied to the evaluation of *in-app* ads due to the following reasons: ***First***, the *in-app targeting* may be based on a number of factors that go beyond what is used for *in-browser* ads, including mobile *apps* installed on the device, the way they are utilised (e.g. heavy gamers may receive specific ads). ***Second***, the classification of ads requires unifying of mobile market place(s) and traditional online environments, as the ads may relate both to merchant websites and to other *apps* that may be purchased and downloaded to the mobile devices. ***Third***, the methodology for collecting information about *in-app* ads is different than for the *in-browser* ads, since the ad delivery process for *in-app* ads changes with every other ad network. ***Finally***, *apps* come with pre-defined *apps* permission to use certain resources, hence, allowing *apps* to filter part of the information to be provided to the ad network.

Figure 6 shows the lifecycle of characterising the ads traffic within the advertising system, both for *in-app* and *in-browser targeted* ads; various data scrapping elements and statistical measures are also shown on the right side of this figure.

Following we discuss few works on the characterisation of *in-app* and *in-browser targeted* ads.

2.7.1 In-app mobile ads

Few studies characterise various features of *in-app* ad traffic with the focus on *targeted* advertising. The MAdScope [33] and [17] collects data from a number of *apps*, probes the ad network to characterise its *targeting* mechanism and reports the *targeted* advertising using profiles of specific interests and preferences. The authors in [32] analyse the ads harvested from 100+ nodes deployed at different geographic locations and 20 Android-based phones and calculated the feasibility of caching and pre-fetching of ads. The authors in [14] characterise the mobile ad traffic from numerous dimensions, such as, the overall traffic, the traffic frequency, and the traffic implications in terms of, using well-known techniques of pre-fetching and caching, energy and network signalling overhead caused by the system. This analysis is based on the data collected from a major European mobile carrier with more than three million subscribers. The [41] shows similar results based on the traces collected from more than 1,700 iPhone and Windows Phone users.

The authors in [42] show that *apps* from the same category share similar data patterns, such as geographic coverage, access time, set of users etc., and follow unique temporal patterns e.g. entertainment *apps* are used more frequently during the night time. The [43] performs a comparative study of the data traffic generated by smartphones and traditional internet in a campus network. Another work [44], studies the cost overhead in terms of the traffic generated by smartphones that is classified into two types of overheads i.e. the portion of the traffic related to the advertisements and the *analytics* traffic i.e. traffic transmitted to the third-party servers for the purpose of collecting data that can be used to analyse users' behaviour etc. Several other works, [45], [46], [47], study *profiling* the energy consumed by smartphone *apps*.

2.7.2 In-browser web ads

There are a number of works on characterising *in-browser* ads with the focus on issues associated with the user privacy [37], [39]. In [6], the authors present classifications of different trackers such as cross-site, in-site, cookie sharing, social media trackers, and demonstrate the dominance of *tracking* for leaking user's privacy, by reverse engineering user's profiles. They further propose a browser extension that helps to protect user's privacy. Prior research works show the extent to which consumers are effectively tracked by third parties and across multiple *apps* [48], mobile devices leaking *Personally Identifiable Information* (*PII*) [49], [50] and *apps* accessing user's private and sensitive information through well defined APIs [51]. Another study [52] reveals by using differential correlation technique in order to identify various *tracking* information used for *targeted* ads. Similarly, [53] investigates the ad fraud that generates spurious revenue affecting the ad agencies. In addition, other studies, such as [54] describes challenges in measuring online ad systems and [40] provides a general understanding of characteristics and changing aspects of advertising and *targeting* mechanisms used by various entities in an ad ecosystem.

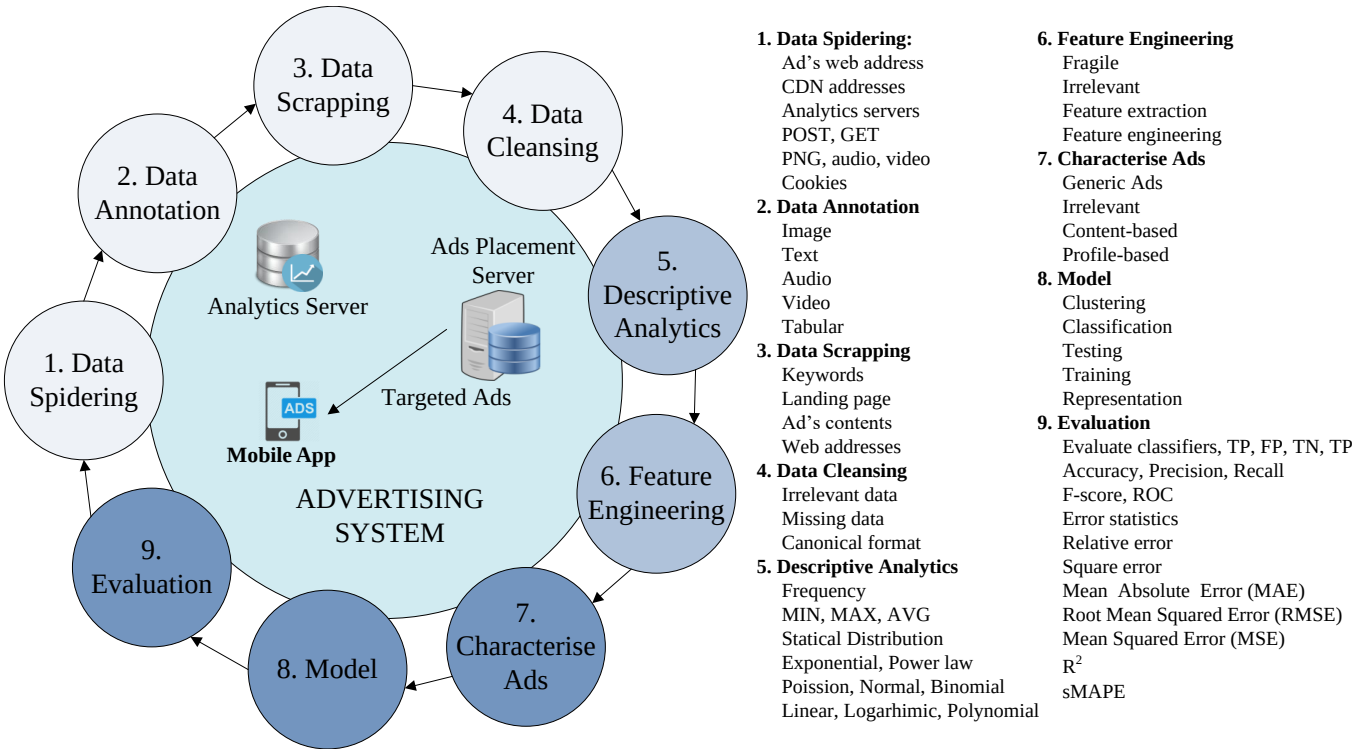

Fig. 6: The process of ads characterisation for both *in-app* and *in-browser targeted* ads.

3 Privacy in Mobile Advertising: Challenges

As mentioned in earlier sections that the *profiling* and *targeted* advertising expose potentially sensitive and damaging information about users, also demonstrated in [55], [56], [57]. There is a growing user awareness of privacy and a number of privacy initiatives, e.g., Apple's enabling of ad blockers in iOS9[9] is representative of a move towards giving users greater control over the display of ads, although applicable only to browser based rather than to *in-app* mobile ads, however, this would greatly affect Google's services, since Google's services are now based on *Web & App* activity[10].

Hence, the purpose of *targeted* advertising is to be able to protect user's privacy and effectively serve relevant ads to appropriate users, in particular, to enable private *profiling* and *targeted* ads without revealing user interests to the adverting companies or third party ad/tracking companies. Furthermore, an private *billing* process to update the advertising network about the ads retrieved/clicked in a privacy preserving manner.

3.1 Privacy attacks

There are various kinds of privacy attacks e.g. in the *direct* attack, the user profile is (legitimately) derived by the *analytics* network (in our previous works [7], [8], [9], we focus on Google AdMob). The *indirect* attack, involves a third party, that monitors the ad traffic (sent in clear text [9], [17] to mobile devices) and infers the user profile based on their *targeted* ads. In both scenarios, the user is not opposed to *profiling* in general and is willing to receive ads on selected topics of interests, but does not wish for specific parts of their profile (*attributes*), based on the usage of *apps* (s)he considers private, to be known to the *analytics* network or any other party, or to be used for ads *targeting*.

3.2 Ad traffic analysis for evaluating privacy leakage

Several works investigate the *in-app* ads traffic primarily for the purpose of privacy and security concerns. The AdRisk [2], an automated tool, analyse 100 *ad libraries* and studies the potential security and privacy leakages of these libraries. The *ad libraries* involve the resource permissions, permission probing and `JavaScript` linkages, and dynamic code loading. Parallel to this work, [58] examines various privacy vulnerabilities in the popular Android-based *ad libraries*. They categorise the permissions required by *ad libraries* into *optional*, *required*, or *un-acknowledged* and investigate privacy concerns such as how user's data is sent in ad requests. The authors in [59] analyse the privacy policy for collecting *in-app* data by *apps* and study various information collected by the *analytics libraries* integrated in mobile *apps*.

Other works [60], [61] study the risks due to the lack of separate working mechanisms between Android *apps*

9. http://au.pcmag.com/mobile-operating-system/31341/opinion/apple-ios-9-ad-blocking-explained-and-why-its-a-ba

10. My Google Activity: https://myactivity.google.com/myactivity?otzr=1

and *ad libraries* and propose methods for splitting their functionality. The authors in [13] monitor the flow of data between the ad services and 250K Android *apps* and demonstrate that currently proposed privacy protecting mechanisms are not effective, since *app* developers and ad companies do not show any concern about user's privacy. They propose a market-aware privacy-enabling framework with the intentions of achieving symmetry between developer's revenue and user's privacy. Another work [62] carried out a longitudinal study in the behaviour of Android *ad libraries*, of 114K free *apps*, concerning the permissions allocated to various *ad libraries* over time. The authors found that over several years, the use of most of the permissions has increased over time raising privacy and security concerns.

There has been several other works, exploring the web advertisements in different ways i.e. form the monetary perspective [21], [63], from the perspective of privacy of information of users [64], from privacy information leakage and to propose methods to protect user data [65], [66], and the E-Commerce [67]. In similar way, a detailed analysis of the web ad networks from the perspective information communicated on network level, the network layer servers, and from the point of the content domains involved in such a system are investigated [68].

3.3 Inference of private information

In recent years, several works [69], [70], [71], [72], [73], [74], [75], [76], [77] have shown that it is possible to infer undisclosed private information of subscribers of online services such as age, gender, relationship status, etc. from their generated contents. The authors in [73] analysed the contents of 71K blogs at blogger.com and were able to accurately infer the gender and age of the bloggers. The authors were able to make their inferences by identifying certain unique features pertaining to an individual's writing style such as parts-of-speech, function words, hyper-links and content such as simple content words and the special classes of words taken from the hand-crafted LIWC (Linguistic Inquiry and Word Count) [78] categories.

Another study [69] has shown that the age demographics of Facebook users (both using *apps* and browsers) can be predicted by analysing the language used in status update messages. Similar inferences have been made for IMDB users based on their movie reviews [74]. Another work [76] predicts age, gender, religion, and political views of users from the queries using models trained from Facebook's 'Like' feature. In [71], the authors analysed client-side browsing history of 250K users and were able to infer various personal attributes including age, gender, race, education and income. Furthermore, a number of studies [79], [80], [81] have demonstrated that sensitive attributes of user populations in online social networks can be inferred based on their social links, group memberships and the privacy policy settings of their friends [82].

3.4 Quantifying privacy algorithms

Quantifying privacy is an important and challenging task as it is important to evaluate the level of privacy protection achieved. It is difficult to formulate a generic metric for quantifying privacy that is applicable to different contexts and due to several types of privacy threats.

For instance, the proposal for fulfilling the privacy requirements using k-anonymity, first proposed in [83], requires that each equivalence class i.e. set of records that are indistinguishable from each other with respect to certain identifying attributes, must have a minimum of k records [84]. Another study [85] reveals that satisfying the privacy requirements for k-anonymity cannot always prevent attribute disclosures mainly for two reasons: First, an attacker can easily discover the sensitive attributes when there is minute diversity in the sensitive attributes, secondly, k-anonymity is not resistant to privacy attacks against the attackers that use background knowledge. They [85] proposes an l-diversity privacy protection mechanism against such attacks and evaluates its practicality both formally and using experiment evaluations. Another work [86] evaluates the limitation of l-diversity and proposes t-closeness, suggesting the distribution of sensitive attributes in an equivalence class must be close to the distribution of attributes in the overall data i.e. distance between two distributions should not be more than the t threshold.

Besides, techniques based on crypto mechanisms, such as PIR, provide privacy protection, for the database present on *single-server*, against the computational complexity [87], [88], *multiple-servers* for protecting privacy against colluding adversaries [89], [90], [91], [92], [93], or protection mechanisms [94] against combined privacy attacks that are either computationally bounded evaluations or against colluding adversaries; these techniques are discussed later in detail in Section 4.

3.5 User information extraction

We experimentally evaluate [9] how to extract user profiles from mobile *analytics* services based on the device identifier of the target; this method was demonstrated using both Google *analytics* and Flurry in the Android environment. Here the user profile, i.e. set of information collected or inferred by the *analytics* services, consists of personally identifiable information such as, unique device ID, demographics, user interests inferred from the *app* usage etc.

An crucial technique to extract user profiles from the *analytics* services (we mainly target Google and Flurry *analytics* services) is to first impersonate the victim's identity; then ***Case 1 Google analytics***: to fetch user profiles from a spoofed device, where the private user profile is simply shown by the Google service as an ads preference setting or ***Case 2 Flurry analytics***: to inject the target's identity into a controlled *analytics app*, which impacts those changes in the Flurry audience analysis report using which the adversary is able to

extract user profile. Following, we first describe how to obtain and spoof a device's identity, subsequently, the user profile extraction for both cases of Google and Flurry is presented in detail.

3.5.1 Information extraction via user profiles from Google

Android system allows users to view and manage their *in-app* ads preferences[11], e.g. to *opt-out* or to *update/delete* interests. This feature retrieves user profile from Google server which is identified by the advertising ID. As a consequence of the device identity spoofing, an adversary is able to access the victim's profile on a spoofed device.

We note that there are at least two possible ways to that an adversary can capture victims Android ID. First, an adversary can intercept the network communication, in order to capture the usage reporting messages sent by third-party tracking APIs, extract the device identifier, and to further use it for ongoing communication with the *analytics* services. Note that it is very easy to monitor IDs of thousands of users in a public hotspots e.g. airport, hospital etc. Similarly, in a confined area, an adversary (e.g. an employer or a colleague) *targeting* a particular individual can even associate the collected device ID to their target (e.g. employees or another colleague). During this privacy attack, we note that Google *analytics library* prevents leakage of device identity by hashing the Android IDs; however it cannot stop other *ad libraries* to transmit such information in plain text (which can be easily be mapped to Google's hashed device ID).

An alternative way, although may be more challenging in practice, is to obtain the target's device identifier from any application (controlled by the adversary) that logs and exports the device's identity information.

3.5.2 Information extraction via user profiles from Flurry

We note that extracting user profiles from Flurry is more challenging since Flurry does not directly allow users to view or edit user's *Interests profiles*. In fact, except the initial consent on the access of device resources, many smartphone users may not be aware of the Flurry's tracking activity.

Figure 7 shows the basic operations of our profile extraction technique within the mobile advertising ecosystem. To compromise a user's private profile, an adversary spoofs the target device, identified by $deviceID_a$, using another Android device or an emulator. Following, the adversary uses a *bespoke app* with a (legitimate) $appID_x$, installed on the *spoofed* device, to trigger a usage report message to Flurry. Accordingly, the *analytics* service is manipulated into believing that $deviceID_a$ is using a new application tracked by the system. Consequently, all user related private information is made

11. Access from `Google Settings` → `Ads` → `Ads by Google` → `Ads Settings`. It claims that Google's ad network shows ads on 2+million non-Google websites and *apps*.

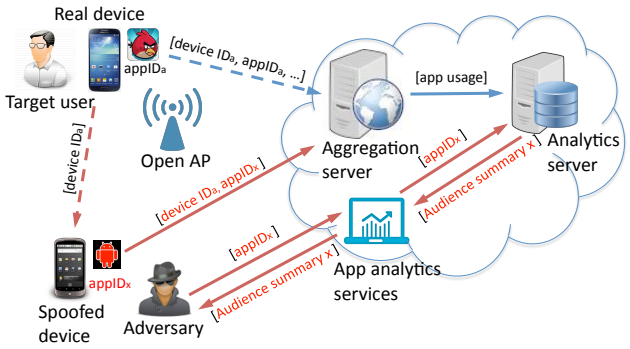

Fig. 7: Privacy leakage attack scenario [9].

accessible to the adversary through audience analysis report of $appID_x$ in Flurry system.

An adversary can easily extract the corresponding statistics and link them to (legitimate) user once the audience report from Flurry targets a unique user. In addition, the adversary will be able to track and access all subsequent changes to the user profile at a later time. In our presented technique, since we do impersonate a particular target's device ID, we can easily associate the target to a 'blank' Flurry-monitored application.

Alternatively, an adversary can derive an individual profile from an aggregated audience analysis report by monitoring report differences before and after a target ID has been spoofed (and as such has been added to the audience pool). Specifically, the adversary has to take a snapshot of the audience analysis report P_t at time t, impersonates a target's identity within his controlled Flurry-tracked application, and then takes another snapshot of the audience analysis report at P_{t+1}. The target's profile is obtained by extracting the difference between P_t and P_{t+1}, i.e. $\Delta(P_t, P_{t+1})$. However in practice, Flurry service updates profile attributes on a weekly basis which means it will take up to a week to extract a full profile per user.

Finally, the *segment* feature provided by Flurry, the *app* audience is further split by applying filters according to e.g. gender, age group and/or developer defined parameter values. This feature allows an adversary to isolate and extract user profiles in a more efficient way. For instance, a possible *segment* filter can be 'only show users who have Android ID value of x' which results in the audience profile containing only one particular user.

3.6 Third-party privacy threats

The third-party A&A libraries have been examined in a number of works, such as [2], [14], [15], [58], [95], which contribute to the understanding of mobile tracking and collecting and disseminating personal information in current mobile networks. The information stored and generated by smartphones, such as call logs, emails, contact list, and GPS locations, is potentially highly sensitive and private to the users. Following, we discuss various means through which users' privacy is exposed.

3.6.1 Third-party tracking

Majority of privacy concerns of smartphone users are because of inadequate access control of resources within the smartphones e.g. Apple iOS and Android, employ fine-grained permission mechanisms to determine the resources that could be accessed by each application. However, smartphone applications rely on users to allow access to these permissions, where users are taking risks by permitting applications with malicious intentions to gain access to confidential data on smartphones [96]. Similarly, privacy threats from collecting individual's online data (i.e. direct and inferred leakage), have been examined extensively in literature, e.g. [10], [97], including third party ad tracking and visiting [98], [99].

Prior research works show the extent to which consumers are effectively tracked by a number of third parties and across multiple *apps* [48], mobile devices leaking *PII* [49], [50], *apps* accessing user's private and sensitive information through well defined APIs [51], inference attacks based on monitoring ads [9] and other data platform such as eXelate[12], BlueKai[13], and AddThis[14] that collect, enrich and resell cookies.

The authors in [100] conducted a user survey and showed that minor number of users pay attention to granting access to permissions during installation and actually understand these permissions. Their results show that 42% of participants were unaware of the existing permission mechanism, only 17% of participant paid attention to permissions during *apps* installation while only 3% of participants fully understood meaning of permissions accessing particular resources. The authors in [2] evaluate potential privacy and security risks of information leakage in *in-app* advertisement by the embedded *libraries* in mobile applications. They studied 100,000 Android *apps* and identified 100 representative *libraries* in 52.1% of *apps*. Their results show that the existing *ad libraries* collect private information that may be used for legitimate *targeting* purposes (i.e., the user location) while other data is harder to justify, such as the users call logs, phone number, browser bookmarks, or even the list of *apps* installed on the phone. Additionally, they identify some *libraries* that use unsafe mechanisms to directly fetch and run code from the Internet, which also leads to serious security risks. A number of works [101], [102], [103], identify the security risks on Android system by disassembling the applications and tracking the flow of various methods defined within various programmed classes.

There are several works to protect privacy by assisting users to manage permissions and resource access. The authors in [104] propose to check the `manifest`[15] files of installed mobile *apps* against the permission assignment policy and blocking those that request certain potentially unsafe permissions. The MockDroid [105] track the resource access and rewrites privacy-sensitive API calls to block information communicated outside the mobile phones. Similarly, the AppFence [106] further improves this approach by adding taint-tracking, hence, allowing more refined permission policies.

3.6.2 Re-identification of sensitive information

Re-identification involves service personalisation based on pervasive spatial and temporal user information that have already been collected e.g. locations that users have already visited. The users are profiled and later on provided with additional offers based on their interests, such as, recommending on places to visit, or people to connect to. There have been a number of research works to identify users based on re-identification technique. For instance, the authors in [107] analyse U.S. Census data and show that on average, every 20 individuals from the dataset share same home or work locations while 5% of people in dataset can be uniquely identified by home-work location pairs. Another related work [108] uniquely identifies US mobile phone users using generalisation technique by generalising the top N homework location pairs. They use location information to derive quasi-identifiers for re-identification of users. Similarly, a number of research works e.g. [109], [110], [111], raise privacy issues in publishing sensitive information and focus on theoretical analysis of *obfuscation* algorithms to protect user privacy.

4 Privacy Preserving Techniques

There are several privacy protection techniques, such as mechanisms based on *differential privacy* i.e. maximising the accuracy of queries from statistical databases while minimising the chances of identifying its records, techniques based on *anonymisation* e.g. encrypting or removing *PII*, *proxy-based* solutions, *k-anonymity* i.e. *generalisation* and *suppression*, *obfuscation* (making the message confusing, willfully ambiguous, or harder to understand), and *crypto-based* techniques such as *private information retrieval* (PIR). Following, we specifically discuss background on *differential privacy* and various PIR techniques and compare these PIR techniques.

4.1 Differential privacy

The concept of *differential privacy*[16] was introduced in [112], a mathematical definition for the privacy loss associated with any released data or *transcript* drawn from

12. https://microsites.nielsen.com/daas-partners/partner/exelate/
13. https://www.oracle.com/corporate/acquisitions/bluekai/
14. https://www.addthis.com/

15. Every Android *app* contains the *manifest* file that describes essential information about app, such as, *app ID*, *app name*, *permission to use device resources used by an app e.g. contacts, camera, list of installed apps etc.*, *hardware and software features the app requires* etc. https://developer.android.com/guide/topics/manifest/manifest-intro.

16. A C++ implementation of *differential privacy* library can be found at https://github.com/google/differential-privacy.

a database. Two datasets D_1 and D_2 differ in at most one element given that one dataset is the subset of the other with larger database contains only one additional row e.g. D_2 can be obtained from D_1 by adding or removing a single user. Hence, a *randomised* function K gives *differential privacy* for the two data sets D_1 and D_2 as: $\mathrm{P}_r\left[K\left(D_1\right) \in S\right] \leq \exp\left(\varepsilon\right) \times \mathrm{P}_r\left[K\left(D_2\right) \in S\right]$. We refer readers to [113] for deeper understanding of *differential privacy* and its algorithms.

Differential privacy is vastly used in the literature for *anonymisation* e.g. a recent initiative to address the privacy concerns by recommending usage of *differential privacy* [114] to illustrate some of the short-comings of direct contact-tracing systems. Google has recently published a *Google COVID-19 Community Mobility Reports*[17] to help public health authorities understand the mobility trends over time across different categories of places, such as retail, recreation, groceries etc., in response to imposed policies aimed at combating COVID-19 pandemic. The authors in [115] use *differential privacy* to publish statistical information of two-dimensional location data to ensure location privacy. Other works, such as [116], [117], partition data dimensions to minimise the amount of noise, and in order to achieve higher privacy accuracy, by using *differential privacy* in response to the given set of queries.

4.2 Private Information Retrieval (PIR)

PIR [88], [89], [94], [118], [119], [120] is a multiparty cryptographic protocol that allows users to retrieve an item from the database without revealing any information to the database server about the retrieved item(s). In one of our previous works [7], our motivation for using PIR rather than other solutions, e.g., Oblivious Transfer [121], [122], is the lower communication and computation overheads of such schemes.

A user wishes to privately retrieve β^{th} record(s) from the database D. D is structured as $r \times s$, where r is the number of records, s the size of each record; s may be divided into words of size w. For *multi-server* PIR, a scheme uses l database servers and has a privacy level of t; k is the number of servers that respond to the client's query, among those, there are v *Byzantine* servers (i.e., malicious servers that respond incorrectly) and h honest servers that send a correct response to the client's query. Following, we briefly discuss and compare various PIR schemes.

4.2.1 Computational PIR (CPIR)

The *single-server* PIR schemes, such as CPIR [87], rely on the computational complexity (under the assumption that an adversary has limited resources) to ensure privacy against malicious adversaries. To privately retrieve the β^{th} record from D, a CPIR client creates a matrix M_β by adding hard noise (based on large disturbance by replacing each diagonal term in M_β by a random bit of 2^{40} words [87]) to the desired record and soft noise (based on small disturbance) to all the other records. The client assumes that the server cannot distinguish between the matrices with hard and soft noises. The server multiplies the query matrix M_β to the database D that results in corresponding response R; the client removes the noise from R to derive the requested record β^{th}.

4.2.2 Recursive CPIR (R-CPIR)

The CPIR mechanism is further improved in terms of communication costs [87] by recursively using the *single-server* CPIR where the database is split into a set of virtual small record sets each considered as a virtual database. The query is hence calculated against part of the database during each recursion. The client recursively queries for the virtual records, each recursion results in a virtual database of smaller virtual records, until it determines a single (actual) record that is finally sent to the client.

4.2.3 Information Theoretic PIR (IT-PIR)

The *multi-server* IT-PIR schemes [89], [90], [91], [92], [93] rely on multiple servers to guarantee privacy against colluding adversaries (that have unbounded processing power) and additionally provide *Byzantine robustness* against malicious servers.

To query a database for β^{th} record with protection against up to t colluding servers, the client first creates a vector e_β, with '1' in the β^{th} position and '0' elsewhere. The client then generates (l, t) *Shamir secret shares* $v_1, v_2, \cdots, v_l$ for e_β. The shares (one each) are subsequently distributed to the servers. Each server i computes the response as $R_i = v_i \cdot D$, this is sent back to the client. The client reconstructs the requested β^{th} record of the database from these responses. The use of of *Shamir secret sharing* enables the recovery of the desired record from (only) $k \leq l$ server responses [89], where $k > t$ (and $t < l$).

4.2.4 Hybrid-PIR (H-PIR)

The *multi-server* H-PIR scheme [94] combines *multi-server* IT-PIR [89] with the recursive nature of the *single-server* CPIR [87] to improve performance, by lowering the computation and communication costs[18]. Let these two schemes be respectively represented by τ for IT-PIR and the γ for the recursive CPIR protocol. A client wants to retrieve β^{th} record then the client must determine the index of virtual records containing the desired records at each step of the recursion until the recursive depth d. The client creates an IT-PIR τ-query for the first index and sends it to each server. It then creates CPIR γ-query during each of the recursive steps and sends it

17. A publicly available resource to see how your community is moving around differently due to COVID-19: http://google.com/covid19/mobility

18. A complete implementation of CPIR, IT-PIR and H-PIR, *Percy++* is present on http://percy.sourceforge.net/.

to all the servers. Similarly, on the server side at each recursive steps; the server splits the database into virtual records each containing actual records, uses the τ server computation algorithm, and finally uses the γ CPIR server computation algorithm. The last recursive step results in the record R_i, that is sent back to the client.

4.3 Comparison of various PIR techniques

Following comparative analysis, based on literature work, would help the selection of various PIR schemes for different applications. We note that various performance metrics relate to the size of query along with the selection of a particular PIR scheme e.g., the CPIR takes longer processing delays and highest bandwidth consumption compared to both the IT-PIR and H-PIR schemes. This is due to the computations involved in query encoding and due to the servers performing *matrix-by-matrix* computations instead of *vector-by-matrix*, as is used by the IT-PIR and H-PIR schemes [94], although, the communication cost can be lowered down using the recursive version of the CPIR [87].

Furthermore, IT-PIR provides some other improvements, such as the *robustness*, which is its ability to retrieve correct records even if some of the servers do not respond or reply with incorrect or malicious responses [93]. It is further evident [94] that both the *single-server* CPIR and the *multi-server* IT-PIR schemes, such as [89], [90], [91], [92], respectively make the assumptions of computationally bounded and that particular thresholds of the servers are not colluding to discover the contents of a client's query. Alternatively, the H-PIR [94], provides improved performance by combining *multi-server* IT-PIR with the recursive nature of *single-server* CPIR schemes respectively to improve the computation and communication costs.

A recent implementation i.e., Heterogeneous PIR [123], enables *multi-server* PIR protocols (implemented using multi-secret sharing algorithm, compatible with *Percy++*[19] PIR library) over non-uniform servers (in a heterogeneous environment where servers are equipped with diverse resources e.g. computational capabilities) that impose different computation and communication overheads. This implementation makes it possible to run PIR over a range of different applications e.g. various resources (ad's contents such as, `JPEG`, `JavaScript` files) present on CDN in distributed environments. Furthermore, this implementation has tested and compared its performance with Goldberg's [89] implementation with different settings e.g., for different database sizes, numbers of queries and for various degrees of heterogeneity. This implementation achieves a trade-off between computation and communication overheads in heterogeneous server implementation by adjusting various parameters.

19. http://percy.sourceforge.net/

4.4 Building Blocks for Enabling PIR

This section introduces various building blocks for enabling PIR techniques i.e. *Shamir secret sharing* and *Byzantine robustness*. It further discusses various techniques that are used for private *billing* i.e. *Threshold BLS signature*, *Polynomial commitment*, and *Zero-knowledge proof* (ZKP).

4.4.1 Shamir secret sharing

The *Shamir secret sharing* [124] scheme divides a *secret* σ into parts, giving each participant e.g. l servers a unique part where some or all of the parts are needed in order to reconstruct the *secret*. If the *secret* is found incorrect then it can be handled through error-correcting codes, such as the one discussed in [125]. Let the σ be an element of some finite field F then the *Shamir scheme* works as follows: a client selects an l distinct non-zero elements $\alpha_1, \alpha_2, \cdots, \alpha_l \in F$ and selects t elements $a_1, a_2, \cdots, a_t \in_R F$ (the $\in_R$ means uniformly at random). A polynomial $f(x) = \sigma + a_1 x + a_2 x^2 + \cdots + a_t x^t$ is constructed and gives the share $(\alpha_i, f(\alpha_i)) \in F \times F$ to the server i for $1 \leq i \leq l$. Now any $t+1$ or more servers can use Lagrange interpolation [93] to reconstruct the polynomial f and, similarly, obtains σ by evaluating $f(0)$.

4.4.2 Byzantine robustness

The problem of *Byzantine* failure allows a server to continue its operation but it incorrectly responds. The *Byzantine* failure may include corrupting of messages, forging messages, or sending conflicting messages through malice or errors. In order to ensure the responses' integrity in a *single-server*, such as PIR-Tor [126], the server can provide a *cryptographic signature* on each database's block. However, in a *multi-server* PIR environment, the main aim of the *Byzantine robustness* is to ensure that the protocol still functions correctly even if some of the servers fail to respond or provide incorrect or malicious responses. The client at the same time might also be interested in figuring out which servers have sent incorrect responses so that they can be avoided in the future.

The *Byzantine robustness* for PIR was first considered by Beimel and Stahl [127], [128]; the scheme called the t-private v-*Byzantine robust* k-out-of-l PIR. The authors take the l-server information-theoretic PIR setting where k of the servers respond, v servers respond incorrectly, and the system can sustain up to t colluding servers without revealing client's query among them. Furthermore, they suggest the *unique decoding* where the protocol always outputs a correct unique block under the conditions $v \leq t \leq k/3$.

The [89] uses the *list decoding*, that is an alternative to unique decoding of error-correcting codes for large error rates, and demonstrates that the privacy level can be substantially increased up to $0 < t < k$ and the protocol can tolerate up to $k - \left\lfloor \sqrt{kt} \right\rfloor - 1$ *Byzantine* servers. Alternatively, the *list decoding* can also be converted to

unique decoding [129] at the cost of slightly increasing the database size [93].

Following schemes are the essential building blocks for enabling private *billing* along with evaluating the PIR techniques for privately retrieving ads from the ad database.

4.4.3 Threshold BLS signature

The *Boneh-Lynn-Shacham* (BLS) [130] is a 'short' *signature* verification scheme that allows a user to verify that the signer is authentic. The signer's private signing key is a random integer $x \in Z_q$ and the corresponding public verification key is $(\hat{g}, \hat{g}^x)$ ($\hat{g}$ is a generator of $\mathbb{G}_2$). The procedure for *signature* verification is as follows: Given the signing key x and a message m, the *signature* is computed via $\sigma = h_x$ where $h = hash(m)$ is a cryptographic hash of m; the verification equation is $e(\sigma, \hat{g}) \stackrel{?}{=} e(h, \hat{g}^x)$, which results in true/false. To fit into scenario of multiple PIR servers; a (k, l)-threshold variant of *BLS signature* can be used where signing keys are the evaluations of a polynomial of degree $(k-l)$ and the master *secret* is the constant term of this polynomial. Similarly, the reconstruction process can be done using Lagrange interpolation. The $(k-l)$ threshold *BLS signature* partly provides the level of *robustness* against the *Byzantine* signers since the *signature* share can be verified independently using the signer's public verification key share.

4.4.4 Polynomial commitment

A *polynomial commitment* [131] scheme allows committers to formulate a constant-sized *commitments* to polynomials that s(he) can commit so that it can be used by a verifier to confirm the stated evaluations of the committed polynomial [132], without revealing any additional information about the committed value(s). An example of the *Polynomial commitment* constructions in [131] provides unconditional hiding if a *commitment* is opened to at most $t-1$ evaluations (i.e. $t-1$ servers for a degree-t polynomial) and provides computational hiding under the discrete $log(DL)$ if polynomial *commitment* is opened to at least t evaluations. As presented in [131], *commitment* to a polynomial $f(x) = a_t x^t + \cdots + a_1 z + a_0$ has the form $\mathcal{C}_f = \left(g^{\alpha^t}\right)^{a_t} \cdots (g^{\alpha})^{a_1} g^{a_0} = g^{f(\alpha)}$ where α is *secret*, $g \in \mathbb{G}_1$ is a generator whose discrete logarithm with respect to g is unknown, including all the bases are part of the *commitment* scheme's *public key*. The verifier, on the other side, can confirm that the claimed evaluations is true by checking if $Ver(\mathcal{C}_f, r, f(r), w) = \left[e(\mathcal{C}_f, \hat{g}) \stackrel{?}{=} e(w, \hat{g}^{\alpha}/\hat{g}^{r}) . e(g, \hat{g})^{f(r)}\right]$ is `true`, here the *commitment* w is called the *witness*; detailed discussion can be found in [131].

4.4.5 Zero-knowledge proof (ZKP)

The *zero knowledge proof* is an interactive protocol between the *prover* and the *verifier* that allows the *prover* to prove to the *verifier* that it holds a given statement without revealing any other information. There are several *ZKPs*, such as range proof to prove that a committed value is non-negative [133], the proof of knowledge of a committed value [134], knowledge proof of a discrete log representation of a number [135], and proof that a *commitment* opens to multiple *commitments* [136]. Besides, there are several batch proof techniques, such as [137], [138] to achieve verification of a basic operation like modular exponentiation in some groups, which significantly reduces the computation time.

4.5 Implementations of private billing for *in-app* mobile advertising

An example implementation of our private *billing* for ads, based on *ZKP* and *Polynomial commitment*, is presented in [7], also shown in 8. In this proposal, we presume that the following information is available to the *client* (software e.g. the AdMob SDK that is integrated in mobile *apps* for requesting ads and tracking user's activity) for all ads in the database: the *Ad* index m, *Ad* category Φ_i, *price tags* C_T^{prs} and C_T^{clk} respectively for *ad presentations* and *ad clicks*, and and the *Advertiser ID* ID_{Adv}. This private *billing* mechanism consists of two parts: the work flow for retrieving ads (Step 1–3) and private *billing* (Step 4–13). In Step 2, the *Ad server* calculates the PIR response and sends it back to the *client*, following, the *client* decodes the PIR response (step 3) and forwards the retrieved ads to the mobile *app*.

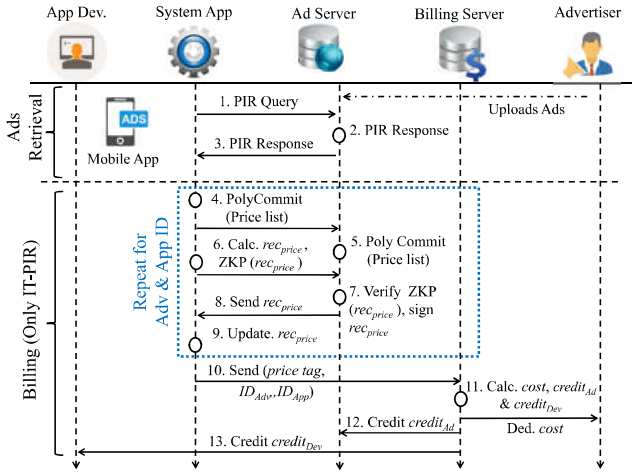

Fig. 8: The work flow for Ads retrieval and billing for *ad presentations* and *ad clicks* [7].

Once the *ads presentation* (or *ad click*) process finishes then it undergoes the *billing* process. The *client* calculates the *receipt* locally, consisting of various components that are used to verify the following: (**a**) price tier for ad presented or ad clicks; (**b**) the ID_{Adv} (used for price deduction from advertiser, as shown in Step 11 of Figure 8); and (**c**) the application ID (helpful for price credit to *App Developer* i.e. Step 13). This *billing* mechanism is based on PS-PIR [90], proposed for *e-commerce*. We note

that this *billing* mechanism is only applicable to single ad requests with no impact on privacy.

As opposed to above implementation, we suggested another proposal [24] for *ad presentations* and *clicks* with the use of *mining* Cryptocurrency (e.g. Bitcoin). The major aim for this proposal was for preserving user privacy, secure payment and for compatibility with the underlying *AdBlock* proposal [24] for mobile advertising system over Blockchain. Following notations are used in this proposal: price tags $C_{prs}^{Ad_{ID}}$ and $C_{clk}^{Ad_{ID}}$ for ad *presentation* and *click*; various *wallets* i.e. *App Developer*'s $wallet_{ID_{APP}}$, *Advertiser*'s $wallet_{AD_{ID}}$, *Billing* server's $wallet_{BS}$; *public-private key* $(PK + /-)$ and (Bitcoin) addresses, i.e. $Add_{ID_{APP}}, Add_{AD_{ID}}, Add_{BS}$. It works as follows: The advertiser buys advertising *airtime*, it signs the message with the amount of Cryptocurrency with her *private key* $(PK-)$, adds *Billing* server's address, requesting a transaction. Following, this request is bind with other transactions and broadcasted over the network for *mining*. Once the transaction completes, the *Billing* server receives its portion of Cryptocurrency in her *wallet*. In addition, the *Miner* initiates *billing* transaction for ads *presentations* or *clicks* respectively by encoding the $C_{prs}^{Ad_{ID}}$ and $C_{clk}^{Ad_{ID}}$ price tags; this amount is then shared with $wallet_{ID_{APP}}$ and $wallet_{AD_{ID}}$ *wallets*.

5 Existing Approaches: Privacy in Mobile Ads: Solutions

The *direct* and *indirect* (i.e., inferred) leakages of individuals' information have raised privacy concerns. A number of research works propose private *profiling* (and advertising) systems [26], [34], [139], [140], [141], [142]. These systems do not reveal either the users' activities or the user's interest profiles to the ad network. Various mechanisms are used to accomplish these goals: Adnostic [26], Privad [140] and Re-priv [139] focus on *targeting* users based on their browsing activities, and are implemented as browser extensions running the *profiling* algorithms locally (in the user's browser). MobiAd [34] proposes a distributed approach, specifically aimed at mobile networks. The use of *differential privacy* is advocated in *Practical Distributed Differential Privacy* (PDDP) [141] and SplitX [142], where differentially private queries are conducted over distributed user data. All these works protect the full user profile and advocate the use of novel mechanisms that necessitate the re-design of some parts or all of the current advertising systems, although some (e.g., Adnostic) can operate in parallel with the existing systems. In addition, the works based on the use of noisy techniques like *differential privacy*, to obfuscate user's preferences may result in a lower accuracy of *targeted* ads (and correspondingly lower revenues), compared to the use of standard *targeting* mechanisms.

Figure 9 shows the lifecycle of proposal for privacy-preserving mobile/web advertising systems; specifically starting from data collection for evaluating privacy/security risks, baseline model and proposed business model for preserving user's privacy, finally model evaluation and its comparison with the baseline model. Various data scrapping elements, statistical measures and privacy preserving techniques are also shown in this figure.

An important thing in the development of private advertising system is that the consumers' trust in privacy of mobile advertising is positively related to their willingness to accept mobile advertising [143], [144]. The AdChoices[20] program (a self-regulation program implemented by the American ad industry), states that consumer could *opt-out* of *targeted* advertising via online choices to control ads from other networks. However, another study [145] examines that the *opt-out* users cause 52% less revenue (and hence presents less relevant ads and lower click through rates) than those users who allow *targeted* advertising. In addition, the authors noted that these ad impressions were only requested by 0.23% of American consumers.

5.1 Private ad ecosystems

There are a number of generic privacy preserving solutions proposed to address the negative impact of ads *targeting*. Anonymity solutions for web browsing include the use of Tor [146], or disabling the use of cookies [147]. These accomplish the goal of preventing user tracking, however, they also prevent any user (profile based) service personalisation, that may actually be a desirable feature for many users despite their privacy concerns.

Research proposals to enable privacy preserving advertising have been more focused on web browsing, as the dominant advertising media e.g., [26], [27], [140], [142], [148], propose to use locally derived user profiles. In particular, Privad [140] and Adnostic [26] use the approach of downloading a wide range of ads from the ad network and locally (in the browser or on the mobile device) selecting ads that match the user's profile. On the other hand, there are a smaller number of works address privacy for mobile *in-app* advertising, with representative works e.g., [7], [8], [24], [28], [34], [149], [150], suggest the *app*-based user *profiling*, stored locally on mobile device. The [7] is based on various mechanisms of PIR and it complements the existing advertising system and is conceptually closest to [149], which uses Oblivious RAM (ORAM) to perform Private Information Retrieval (PIR) on a secure coprocessor hardware. However, unlike our solution it relies on specific (secure) hardware to enable PIR, which may limit its applicability in a general setting.

5.2 Privacy techniques and their usage in targeted advertising systems

Various proposals address selected aspects of privacy in advertising, described in Sections 3 and 4: collection of data belonging to identified individuals, *profiling* of

20. https://optout.aboutads.info/?c=2&lang=EN

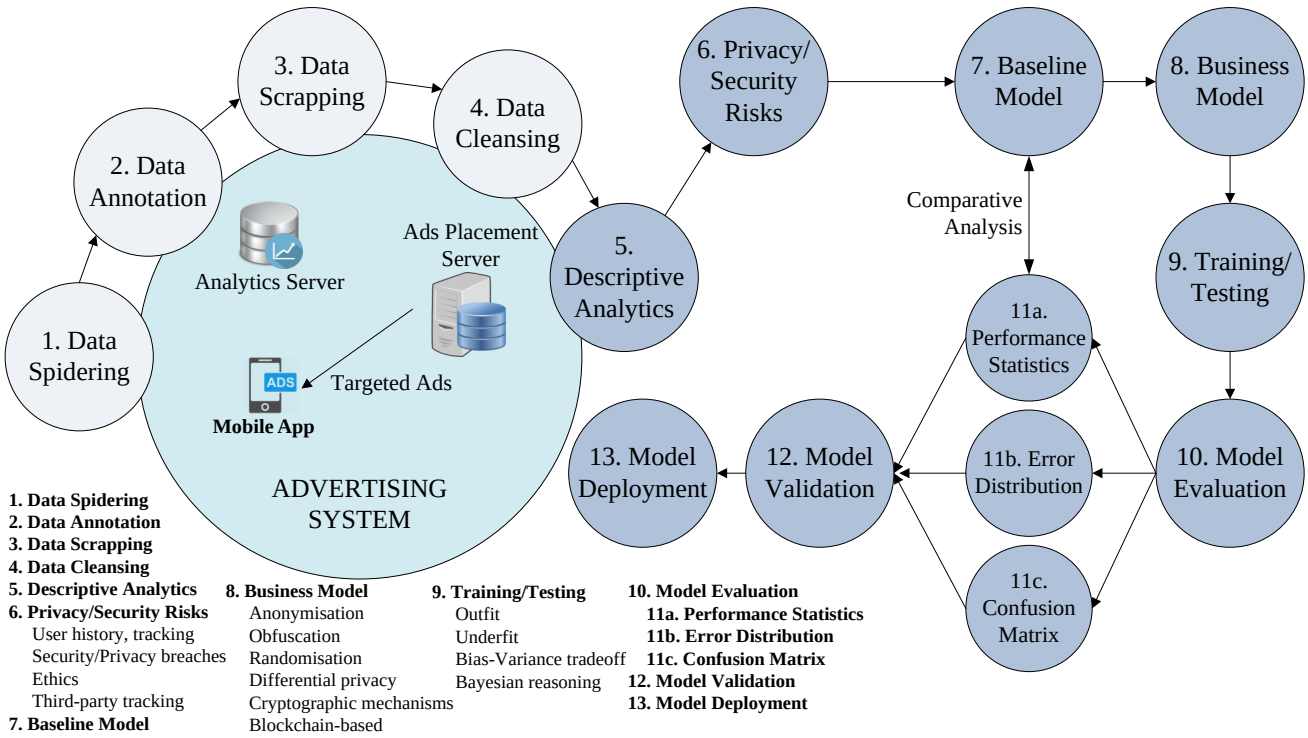

Fig. 9: Lifecycle of proposal for privacy-preserving advertising systems for both *in-app* and *in-browser targeted* ads.

those individuals, distribution of *targeted* ads in line with user's profiles and *accounting* and *billing* for ads (including ad impressions and ad clicks). These include: mechanisms to prevent the identification of users and therefore disable monitoring of individual user activity, methods to provide user *profiling* in a privacy preserving way, and complete systems that suggest changes to all ad related functionality while respecting user's privacy. Following we present various privacy-preserving advertising systems based on different privacy techniques.

5.2.1 Anonymisation

The simplest and most straightforward way to *anonymise* data includes masking or removing data fields (attributes) that comprise *PII*. These include direct identifiers like names and addresses, and quasi-identifiers (QIDs) such as gender and zip code, or an IP address; the later can be used to uniquely identify individuals. It is assumed that the remainder of the information is not identifying and therefore not a threat to privacy (although it contains information about individuals, e.g. their interests, shopping patterns, etc.). A second approach is to generalise QIDs, e.g., by grouping them into a higher hierarchical category (e.g., locations into post codes); this can also be accomplished according to specified *generalisation* rules. *Anonymisation* mechanisms that deal with selected QIDs according to predetermined rules include k-anonymity [151] and it's variants like l-diversity [85] and t-closeness [86]. These, in their simplest form (k-anonymity) modify (*generalise*) individual user records so that they can be grouped into identical (and therefore indistinguishable) groups of k, or additionally apply more complex rules (l-diversity and t-closeness).

A number of proposals advocate the use of locally (either in the browser of the mobile device) derived user profiles, where user's interests are *generalised* and/or partially removed (according to user's privacy preferences), before being forwarded to the server or an intermediary that selected the appropriate ads to be forwarded to the clients. In the context of *targeted* advertising, the removal of direct identifiers includes user IDs (replacing them with temporary IDs) or mechanisms to hide used network address (e.g., using TOR [146]). However, if only the most obvious *anonymisation* is applied without introducing additional (*profiling* and *targeting* oriented) features, the ad networks ecosystem would be effectively disabled. Therefore, we only mention representative solutions from this category and concentrate on the privacy-preserving mechanisms that enable *targeted* ads.

The privacy requirements are also, in a number of prior works, considered in parallel with achieving bandwidth efficiency for ad delivery, by using caching mechanisms [32], [34], [140]. Furthermore, such techniques have been demonstrated to be vulnerable to composition attacks [152], and can be reversed (with individual users identified) when auxiliary information is available (e.g. from online social networks or other publicly available sources) [153], [154].

In Adnostic [26], each time a webpage (containing ads) is visited by the user; the client software receives a set of

generic ads, randomly chosen by the broker. The most appropriate ads are then selected locally, by the client, for presentation to the user; this is based on the locally stored user profile. We have categorised this work as a *generalisation* mechanism as the served ads are generic (non-personalised), although it could arguably be considered under the *randomisation* techniques. Adnostic also uses crypto mechanisms, as detailed in sub-section 4.2. We note that in [26] the user's privacy (visited pages or ad clicks) is not protected from the broker.

In Privad [25], [140], a local, (detailed) user profile is generated by the Privad client and then *generalised* before sending to the ads broker in the process of requesting (broadly) relevant ads. All communication with the broker is done through the dealer, which effectively performs the functions of an *anonymising* proxy; the additional protection is delivered by encrypting all traffic, this protecting user's privacy from the dealer. The proposed system also includes monitoring of the client software to detect whether any information is sent to the broker using, e.g., a covert channel. Similarly, in MobiAd [34], the authors propose a combination of peer-to-peer mechanisms that aggregates information from users and only presents the aggregate (*generalised* activity) to the ad provider, for both ad impressions and clicks. Caching is utilised to improve efficiency and Delay tolerant networking for forwarding the information to the ad network. Similarly, another work [155] proposes combining of users interests via an ad-hoc network, before sending them to the ad server.

Additionally, some system proposals [156] advocate the use of *anonymisation* techniques (*l*-diversity) in the *targeting* stage, where the ads are distributed to users, while utilising alternative mechanisms for *profiling*, learning and statistics gathering.

5.2.2 Obfuscation

A recent work [157] carries out a large scale investigation of *obfuscation* use where authors analyse 1.7 million free Android *apps* from Google Play Store to detect various *obfuscation* techniques, finding that only 24.92% of apps are obfuscated by the developer. There are several *obfuscation* mechanisms for protecting private information, such as the *obfuscation* method presented in [158] that evaluates different classifiers and *obfuscation* methods including greedy, sampled and random choices of obfuscating items. They evaluate the impact of *obfuscation*, assuming prior knowledge of the classifiers used for the inference attacks, on the utility of recommendations in a movie recommender system. A practical approach to achieving privacy [159], which is based on the theoretical framework presented in [160], is to distort the view of the data before making it publicly available while guaranteeing the utility of the data. Similarly, [161] proposes an algorithm for publishing partial data that is safe against the malicious attacks where an adversary can do the inference attacks using association rule in publicly published data.

Another work, 'ProfileGuard' [28] and its extension [8] propose an *app*-based profile *obfuscation* mechanism with the objective of eliminating the dominance of private interest categories (i.e. the prevailing private interest categories present in a user profile). The authors provide insights to Google AdMob *profiling* rules, such as showing how individual *apps* map to user's interests within their profile in a deterministic way and that AdMob requires a certain level of activity to build a *stable* user profile. These works use a wide-range of experimental evaluation of Android *apps* and suggest various *obfuscation* mechanisms e.g. *similarity* with user's existing *apps*, *bespoke* (customised to profile *obfuscation*) and *bespoke++* (*resource-aware*) strategies. Furthermore, the authors also implement a POC 'ProfileGuard' *app* to demonstrate the feasibility of an automated *obfuscation* mechanism.

Following, we provide an overview of prior work in both *randomisation* (generic noisy techniques) and *differentially private* mechanisms.

5.2.3 Randomisation

In the *randomisation* methods, noise is added to distort user's data. Noise can either be added to data values (e.g., movie ratings or location GPS coordinates), or, more applicable to *profiling* and user *targeting*, noise is in the form of new data (e.g., additional websites that the user would not have visited normally are generated by a browser extension [162]), added in order to mask the true vales of the records (browsing history). We note that [162] protects the privacy of user's browsing interests but does not allow (privacy preserving) *profiling* or selection of appropriate *targeted* ads.

The idea behind noise addition is that specific information about user's activities can no longer be recovered, while the aggregate data still contains sufficient statistical accuracy so that it can be useful for analysis (e.g., of trends). A large body of research work focuses on generic noisy techniques e.g. [163] proposed the approach of adding random values to data, generated independently of the data itself, from a known e.g., the uniform distribution. Subsequent publications (e.g., [164]) improve the initial technique, however other research work [165] has identified the shortcomings of this approach, where the added noise may be removed by data analysis and the original data (values) recovered.

A novel noisy technique for privacy preserving personalisation of web searches was also recently proposed [166]. In this work, the authors use 'Bloom' cookies that comprise a noisy version of the locally derived profile. This version is generated by using Bloom filters [167], an efficient data structure; they evaluate the privacy versus personalisation trade-off.

5.2.4 Differential privacy

Differential privacy [168] work has, in recent years, resulted in a number of system works that advocate the practicality of this, previously predominantly theoretical

research field. The authors in [141] propose a system for *differentially private* statistical queries by a data aggregator, over distributed users data. A proxy (assumed to be *honest-but-curious*) is placed between the analyst (aggregator) and the clients and secure communications including authentication and traffic confidentiality are accomplished using TLS [169]. The authors also use a cryptography solution to provide additional privacy guarantees.The SplitX system [142] also provides *differential privacy* guarantees and relies on intermediate nodes, which forward and process the messages between the client that locally stores their (own) data and the data aggregator. Further examples include works proposing the use of distributed *differential privacy* [170] and [171].

5.2.5 Cryptographic mechanisms

A number of different cryptographic mechanisms have been proposed in the context of *profiling* and *targeted* advertising (or, more broadly, search engines and recommender systems): Private Information retrieval (PIR) [88], [89], [94], [118], [119], [120], homomorphic encryption [172], zero knowledge proofs [133], [134], [135], [136] and mixing [173].

A web-based advertising system based on Private Information retrieval was first proposed by Juels [174] , where they use *information-theoretic* (threshold) PIR in an *honest-but-curious multi-server* architecture. Central to their system is the choice of a negotiant function, that is used by the advertiser to select ads, starting from a user's profile - the authors describe both a semi-private and a fully private *information-theoretic* (threshold) PIR in an *honest-but-curious multi-server* architecture. They evaluate the benefits of both alternatives in regards to security, computational cost and communication overheads.

The ObliviAd proposal [149] uses a PIR solution based on bespoke hardware (secure coprocessor), which enables on-the-fly retrieval of ads. The authors propose the use of Oblivious RAM (ORAM) model, where the processor is a "black box", with all internal operations, storage and processor state being unobservable externally. ORAM storage data structure comprises of entries that include a combination of keyword and a corresponding ad (multiple ads result in multiple entries). The accounting and *billing* are secured via the use of using electronic tokens (and mixing [175], [176]). More generally, a system that enables private e-commerce using PIR was investigated in [90], with tiered pricing with record level granularity supported via the use of the proposed Priced Symmetric PIR (PS-PIR) scheme. Multiple sellers and distributed accounting and *billing* are also supported by the system.

Additionally, cryptographic solutions can be used to provide part of the system functionality. They are commonly used in conjunction with *obfuscation*, e.g., in [170], [171] or *generalisation* [26]. Adnostic [26] uses a combination of homomorphic encryption and zero-knowledge proof mechanisms to enable accounting and *billing* in the advertising system in a (for the user) privacy preserving way. Effectively, the user is protected as neither the publisher (website that includes the ads) or the advertisers (that own the ads) have knowledge about which users viewed specific ads. The authors in [170] also combine *differential privacy* with a homomorphic cryptosystem, to achieve privacy in a more generic setting of private data aggregation of distributed data. Similarly, Shi et al. [171] also use a version of homomorphic techniques to enable private computing of sums based on distributed time-series data by an un-trusted aggregator.

Chen et al. [141] uses cryptographic mechanism to combine client-provided data (modified in accordance with *differential privacy*). They utilise a probabilistic Goldwasser-Micali cryptosystem [177]. In their subsequent work [142], the authors use an XOR-based cryptomechanism to provide both anonymity and unlinkability to analysis (queries) of *differentially private* data distributed on user's devices (clients). A cryptography technique, mixing [175], [176] is also commonly used as part of *anonymisation* [149], [174], where mix servers are used as intermediaries that permute (and re-encrypt) the input.

5.2.6 Blockchain-based advertising systems

Blockchain [178] has numerous applications and has been widely used, e.g. IoT [179], Bid Data [180], Healthcare [181], Banking and finance [182] etc. Blockchain has become a new foundation for decentralised business models, hence in the environment of advertising platform, made it a perfect choice for restricting communication between mobile *apps* (which is potentially a big source of private data leakage) and the ad/analytics companies and keeping individual's privacy.

To our knowledge, we note that there are very limited works available for Blockchain-based mobile *targeted* ads in the literature e.g. the [29] presents a decentralised *targeted* mobile coupon delivery scheme based on Blockchain. The authors in this work match the behavioral profiles that satisfy the criteria for *targeting* profile, defined by the vendor, with relevant advertisements. However, we note that this framework does not include all the components of an advertising system including user profiles construction, detailed structure of various Blockchain-based transactions and operations, or other entities such as *Miner* and the *billing* process. Our recent work, *AdBlock* [24], presents a detailed framework (in addition to Android-based POC implementation i.e. a *Bespoke Miner*) for privacy preserving user *profiling*, privately requesting ads, the *billing* mechanisms for presented and clicked ads, mechanism for uploading ads to the cloud, various types of transactions to enable advertising operations in Blockchain-based network, and methods for *access policy* for accessing various resources, such as accessing ads, storing mobile user profiles etc. This framework is parented in Figure 10. We further experimentally evaluate its applicability by implementing various critical components: evaluating user profiles, implementing *access policies*, encryption and decryption of

user profiles. We observe that the processing delays with various operations evaluate to an acceptable amount of processing time as that of the currently implemented ad systems, also verified in [7].

Summary of various privacy preserving approaches, in terms of *architecture*, *mechanism*, *deployment* and *app domain*, for both *in-browser* web and *in-app* mobile advertising systems is given in Table 1.

5.3 The economic aspects of privacy

Research works also investigate the notion of compensating users for their privacy loss, rather than imposing limits on the collection and use of personal information.

Ghosh and Roth [186] studied a market for private data, using *differential privacy* as a measure of the privacy loss. The authors in [187] introduce transactional privacy, which enables the users to sell (or lease) selected personal information via an auction system. On a related topic of content personalisation and *in-browser* privacy, in RePriv [139] the authors propose a system that fits into the concept of a marketplace for private information. Their system enables controlling the level of shared (local) user profile information with the advertising networks, or, more broadly, with any online entity that aims to personalise content.

6 Open Research Issues

In this section, we present various future research directions that require further attention from the research community i.e. diffusion of user data in Real Time Bidding (RTB) scenarios and associated privacy risks, the complicated operations of advertising system, the user-driven private mobile advertising systems and its private *billing* mechanism.

6.1 Diffusion of user tracking data

A recent shift in the online advertising has enabled by the advertising ecosystem to move from ad networks towards ad exchanges, where the advertisers bid on impressions being sold in RTB auctions. As a result, the A&A companies closely collaborate for exchanging user data and facilitate bidding on ad impressions and clicks [188], [189]. In addition, the RTB cause A&A companies to perform additional tasks of working with publishers to help manage their relationship for ad exchange (in addition to user's tracking data) and to optimise the ad placement (i.e. *targeted* ads) and bidding on advertiser's behalf. This has made the online advertising operations and the advertising ecosystems themselves extremely complex.

Hence, it is important for the A&A companies to model (in order to accurately capture the relationship between publisher and A&A companies) and evaluate the impact of RTB on the diffusion of user tracking (sensitive) data. This further requires assessing the advertising impact on the user's contexts and *profiling* interests, which is extremely important for its applicability and scalability in the advertising scenarios. This will also help the A&A companies and publisher to effectively predict the tracker domain and to estimate their advertising revenue. Furthermore, to ensure the privacy of user data since the data is collected and disseminated in a distributed fashion i.e. users affiliated to different *analytics* and advertising platforms and shared their data across diverse publishers. This also necessitates a distributed platform for the efficient management and sharing of distributed data among various A&A platforms and publishers. In particular, the RTB has demanded to develop efficient methods for distributed and private data management.

6.2 Complex operations of advertising system

The complexity of online advertising poses various challenges to user privacy, processing-intensive activities, interactions with various entities (such as CDN, *analytics* servers, etc.) and their tracking capabilities. In order to reduce the complexity of the advertising systems, we envision few more areas of research: devising processing-sensitive frameworks, limiting the direction-redirection of requests among A&A entities, unveil user data exchange processes within the ad platform, identifying new privacy threats and devising new protection mechanisms. Unveiling user data exchange will expose the extent to which the intermediate entities prone to adversarial attacks. Hence, it requires a better knowledge of adversary, which will contribute to develop protection mechanisms for various kinds of privacy threats, such as, interest-based attacks, direct privacy attacks. Note that this will further require comparative analysis of basic and new proposals for the trade-off achieved between privacy and computing overheads of processing user's ad retrieval requests/responses, communication bandwidth consumption and battery consumption.

6.3 Private user-driven mobile advertising systems

An enhanced user-driven private advertising platform is required where the user interest (vis-à-vis their privacy) and advertising system's business interests may vary, in addition, the assessment of user information as an inherent economic value will help to study the tradeoff between such values and user privacy within the advertising system. This will require the proposal for complex machine learning techniques to enhance ads *targeting* (since previous works found that majority of received ads were not tailored to intended user profiles [17], [33], which will ultimately help advertising systems to increase their revenues and enhance user experience in receiving relevant ads. Likewise, introducing novel privacy preserving mechanisms, a very basic step would be to combine various proposals, as described in Section 5, which will introduce more robust and useful privacy solutions for various purposes: enhanced user *targeting*, invasive tracking behaviors, better adapting

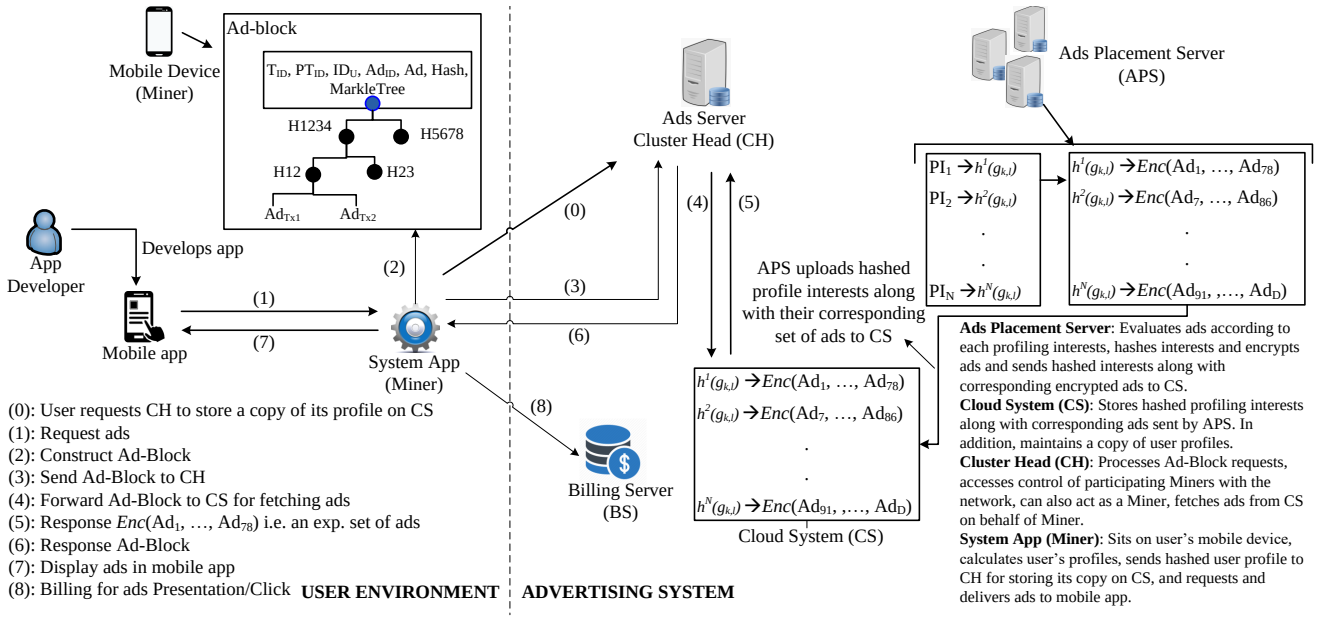

Fig. 10: A framework for secure user *profiling* and Blockchain-based *targeted* advertising system for *in-app* mobile ads [24]. Description of various operation redirections (left side) and advertising entities (right side) is also give in this figure.

Ref	Architecture	Mechanism	Deployment	Domain
Privad [140]	3rd-party anonymising proxy	Crypto	Browser add-on	Web
Adnostic [26]	Complements to existing sys	Crypto billing	Firefox extension	
PASTE [170]	Untrusted third party	Fourier Perturbation Algo	Browser add-on	
[183]	Cookie management	User preference	Standalone	
[184]	Anonymising proxy	Differential privacy		
DNT [185][21]	Delay Tolerant Network	HTTP header	Browser side	
MobiAd [34]		Encryption	Mobile phone	Mobile
ObliviAd [149]	Complements existing sys	Crypto-based	Client/Server sides	
[150]		Differential privacy		
SplitX [142]		XOR-based encryption		
CAMEO [32]		Context prediction		
ProfileGuard [8], [28]		Profile Obfuscation		
[29]		Blockchain		
AdBlock [24]				
[7]	Autonomous system	Crypto-based	Standalone	

TABLE 1: Summary of the *in-browser* web and *in-app* mobile advertising systems.

privacy enhancing technologies, better adapt the changing economic aspects and *ethics* in ads *targeting*. Another research direction would be to extend the analysis of privacy protection mechanisms to other different players, such as, advertisers, ad exchange, publishers with the aim to analyse and evaluate privacy policies and protection mechanisms that are claimed by these parties. This would help various entities in the advertising system to identify the flaws and further improve their working environment.

Another research direction would be to create smarter privacy protection tools on the user side i.e. to create such tools as an essential component of mobile/browser-based platform within the advertising ecosystem. To develop such tools where users effectively enforce various protection strategies, it require various important parameters of usability, flexibility, scalability etc., to be considered to give users transparency and control over their private data.

Another research direction would be to extend the analysis of privacy protection mechanisms to other different players, such as, advertisers, ad exchange, publishers with the aim to analyse and evaluate privacy policies and protection mechanisms that are claimed by these parties. This would help various entities in the advertising system to identify the flaws and further improve their working environment.

6.4 Privet billing mechanism

Billing for both *ad presentations* and *clicks* is an important component of online advertising system. As discussed in Section 4.4, a private *billing* proposal is based on *Threshold BLS signature*, *Polynomial commitment*, and *Zero knowl-*

21. It [185] proposes a DNT field in the HTTP header that requests a web application to either disable the tracking (where it is automatically set) or cross-site the user tracking of an individual user.

edge proof (ZKP), which are based on PIR mechanisms and *Shamir secret sharing* scheme along with *Byzantine robustness*. The applicability of this private *billing* model can be verified in the online advertising system, which would require changes on both the user and ad system side. Furthermore, note that the this private *billing* mechanism, implemented via *polynomial commitment* and *zero-knowledge proof*, is highly resource consuming process, henceforth, an alternative implementation with reduced processing time and query request size can be achieved via implementing together *billing* with PIR using *multi-secret sharing* scheme. In addition, to explore the effect of *multi-secret sharing* scheme in multiple-server PIR and hence comparative analysis to choose between the two variations of *single-secret* and *multi-secret sharing* system implementations. *Multi-secret sharing* scheme would help reduce the communication bandwidth and delays along with the processing time of query requests/responses

In addition, our *billing* mechanism for *ad presentations* and *clicks* presented in [7], also described in Section 4.5, is applicable only to single ad requests with no impact on privacy. However, the broader parameter values (simultaneously processing multiple ad requests) and the use of other PIR techniques, such as Hybrid-PIR [94] and Heterogeneous-PIR [123], can be used to efficiently make use of processing time.

Furthermore, with the rise in popularity of Cryptocurrencies, many businesses and individuals have started investing in them, henceforth, the applicability of embedding the Cryptocurrency with the existing *billing* methods needs an investigation and developing new frameworks for coexisting the *billing* payments with the Cryptocurrency market. In addition, this would require techniques for purchasing, selling, and transferring Cryptocurrency among various parties i.e. ad systems, *app* developers, publishers, advertisers, crypto-markets, and miners. A further analysis would require investigating the impact of such proposals on the current advertising business model with/without a significant effect.

An important research direction is to explore implementation of private advertising systems in Blockchain networks since there is limited Blockchain-based advertising systems e.g., [24], [29]. The [24] presents the design of a decentralised framework for *targeted* ads that enables private delivery of ads to users whose behavioral profiles accurately match the presented ads, defined by the advertising systems. This framework provides: a private *profiling* mechanism, privately requesting ads from the advertising system, the *billing* mechanisms for ads monetisation, uploading ads to the cloud system, various types of transactions to enable advertising operations in Blockchain-based network, and *access policy* over cloud system for accessing various resources (such as ads, mobile user profiles). However, its applicability in an actual environment is still questionable, in addition to, the coexistence of *ads-billing* mechanism with Cryptocurrency.

7 Conclusion

Targeted/Online advertising has become ubiquitous on the internet, which has triggered the creation of new internet ecosystems whose intermediate components have access to billions of users and to their private data. The lack of transparency of online advertising, the A&A companies and their operations have posed serious risks to user privacy. In this article, we break down the various instances of *targeted* advertising, their advanced and intrusive tracking capabilities, the privacy risks from the information flow among various advertising platforms and ad/analytics companies, the *profiling* process based on user's private data and the *targeted* ads delivery process. Several solutions have been offered in the literature to help protect user privacy in such a complex ecosystem, henceforth, we provide a wide range of mechanisms that were classified based on the privacy mechanisms used, ad serving paradigm and the deployment scenarios (browser and mobile). Some of the solutions are very popular among internet users, such as blocking, however their blocking mechanism negatively impacts the advertising systems. On the other hand, majority of the proposals provide naive privacy that require a lot of efforts from the users; similarly, other solutions demand structural changes with the advertising ecosystems. We have found that it is very hard, based on various privacy preserving approaches, while demanding for devising novel approaches, to provide user privacy that could give users more control over their private data and to reduce the financial impact of new systems without significantly changing the advertising ecosystems and their operations.

References

[1] buildfire, "Mobile app download and usage statistics (2020)," *https://buildfire.com/app-statistics/*.

[2] M. C. Grace, W. Zhou, X. Jiang, and A.-R. Sadeghi, "Unsafe exposure analysis of mobile in-app advertisements," pp. 101–112, 2012.

[3] W. Enck, D. Octeau, P. McDaniel, and S. Chaudhuri, "A study of android application security.," vol. 2, p. 2, 2011.

[4] T. Book and D. S. Wallach, "A case of collusion: A study of the interface between ad libraries and their apps," pp. 79–86, 2013.

[5] A. Chaabane, G. Acs, and M. A. Kaafar, "You are what you like! information leakage through users' interests," 2012.

[6] C. Castelluccia, M.-A. Kaafar, and M.-D. Tran, "Betrayed by your ads!," pp. 1–17, 2012.

[7] I. Ullah, B. G. Sarwar, R. Boreli, S. S. Kanhere, S. Katzenbeisser, and M. Hollick, "Enabling privacy preserving mobile advertising via private information retrieval," in *2017 IEEE 42nd Conference on Local Computer Networks (LCN)*, pp. 347–355, IEEE, 2017.

[8] I. Ullah, R. Boreli, S. S. Kanhere, S. Chawla, T. A. Ahanger, and U. Tariq, "Protecting private attributes in app based mobile user profiling," *IEEE Access*, vol. 8, pp. 143818–143836, 2020.

[9] T. Chen, I. Ullah, M. A. Kaafar, and R. Boreli, "Information leakage through mobile analytics services," in *15th International Workshop on Mobile Computing Systems and Applications*, ACM HotMobile, 2014.

[10] S. Mamais, *Privacy-preserving and fraud-resistant targeted advertising for mobile devices*. PhD thesis, Cardiff University, 2019.

[11] Y. Liu and A. Simpson, "Privacy-preserving targeted mobile advertising: requirements, design and a prototype implementation," *Software: Practice and Experience*, vol. 46, no. 12, pp. 1657–1684, 2016.

[12] Y. Wang, E. Genc, and G. Peng, "Aiming the mobile targets in a cross-cultural context: Effects of trust, privacy concerns, and attitude," *International Journal of Human–Computer Interaction*, vol. 36, no. 3, pp. 227–238, 2020.

[13] I. Leontiadis, C. Efstratiou, M. Picone, and C. Mascolo, "Don't kill my ads!: balancing privacy in an ad-supported mobile application market," p. 2, 2012.

[14] N. Vallina-Rodriguez, J. Shah, A. Finamore, Y. Grunenberger, K. Papagiannaki, H. Haddadi, and J. Crowcroft, "Breaking for commercials: characterizing mobile advertising," pp. 343–356, 2012.

[15] S. Han, J. Jung, and D. Wetherall, "A study of third-party tracking by mobile apps in the wild," 2012.

[16] "Flurry advertisers, publishers, and analytics," *www.flurry.com*, 2016.

[17] I. Ullah, R. Boreli, M. A. Kaafar, and S. S. Kanhere, "Characterising user targeting for in-app mobile ads," pp. 547–552, 2014.

[18] C. Tracking, "Understanding conversion tracking," 2020.

[19] V. Ng and M. K. Ho, "An intelligent agent for web advertisements," *International Journal of Foundations of Computer Science*, vol. 13, no. 04, pp. 531–554, 2002.

[20] A. Thawani, S. Gopalan, and V. Sridhar, "Event driven semantics based ad selection," vol. 3, pp. 1875–1878, 2004.

[21] J. Yan, N. Liu, G. Wang, W. Zhang, Y. Jiang, and Z. Chen, "How much can behavioral targeting help online advertising?," pp. 261–270, 2009.

[22] J. Jaworska and M. Sydow, "Behavioural targeting in on-line advertising: An empirical study," in *Web Information Systems Engineering-WISE 2008*, pp. 62–76, Springer, 2008.

[23] J. Shin and J. Yu, "Targeted advertising: How do consumers make inferences?," 2019.

[24] I. Ullah, S. S. Kanhere, and R. Boreli, "Privacy-preserving targeted mobile advertising: A blockchain-based framework for mobile ads," *arXiv preprint arXiv:2008.10479*, 2020.

[25] S. Guha, B. Cheng, A. Reznichenko, H. Haddadi, and P. Francis, "Privad: Rearchitecting online advertising for privacy," *Proceedings of Hot Topics in Networking (HotNets)*, 2009.

[26] V. Toubiana, A. Narayanan, D. Boneh, H. Nissenbaum, and S. Barocas, "Adnostic: Privacy preserving targeted advertising," 2010.

[27] O. Rafieian and H. Yoganarasimhan, "Targeting and privacy in mobile advertising," *Available at SSRN 3163806*, 2020.

[28] I. Ullah, R. Boreli, S. S. Kanhere, and S. Chawla, "Profileguard: Privacy preserving obfuscation for mobile user profiles," pp. 83–92, 2014.

[29] Y. Gu, X. Gui, P. Xu, R. Gui, Y. Zhao, and W. Liu, "A secure and targeted mobile coupon delivery scheme using blockchain," in *International Conference on Algorithms and Architectures for Parallel Processing*, pp. 538–548, Springer, 2018.

[30] T. Trzcinski, "Analyse, target & advertise privacy in mobile ads,"

[31] "Mobile advertising market size, share & industry analysis, forecast 2019-2026," *https://www.fortunebusinessinsights.com/mobile-advertising-market-102496*, Accessed on June, 2020.

[32] A. J. Khan, K. Jayarajah, D. Han, A. Misra, R. Balan, and S. Seshan, "Cameo: A middleware for mobile advertisement delivery," pp. 125–138, 2013.

[33] S. Nath, "Madscope: Characterizing mobile in-app targeted ads," pp. 59–73, 2015.

[34] H. Haddadi, P. Hui, and I. Brown, "Mobiad: private and scalable mobile advertising," pp. 33–38, 2010.

[35] R. Balebako, P. Leon, R. Shay, B. Ur, Y. Wang, and L. Cranor, "Measuring the effectiveness of privacy tools for limiting behavioral advertising," 2012.

[36] C. E. Wills and C. Tatar, "Understanding what they do with what they know," pp. 13–18, 2012.

[37] A. Goldfarb and C. Tucker, "Online display advertising: Targeting and obtrusiveness," *Marketing Science*, vol. 30, no. 3, pp. 389–404, 2011.

[38] A. Farahat and M. C. Bailey, "How effective is targeted advertising?," pp. 111–120, 2012.

[39] D. S. Evans, "The online advertising industry: Economics, evolution, and privacy," *Journal of Economic Perspectives, Forthcoming*, 2009.

[40] P. Barford, I. Canadi, D. Krushevskaja, Q. Ma, and S. Muthukrishnan, "Adscape: Harvesting and analyzing online display ads," pp. 597–608, 2014.

[41] P. Mohan, S. Nath, and O. Riva, "Prefetching mobile ads: Can advertising systems afford it?," pp. 267–280, 2013.

[42] Q. Xu, J. Erman, A. Gerber, Z. Mao, J. Pang, and S. Venkataraman, "Identifying diverse usage behaviors of smartphone apps," pp. 329–344, 2011.

[43] S.-W. Lee, J.-S. Park, H.-S. Lee, and M.-S. Kim, "A study on smart-phone traffic analysis," pp. 1–7, 2011.

[44] L. Zhang, D. Gupta, and P. Mohapatra, "How expensive are free smartphone apps?," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 16, no. 3, pp. 21–32, 2012.

[45] A. Pathak, Y. C. Hu, and M. Zhang, "Where is the energy spent inside my app?: fine grained energy accounting on smartphones with eprof," pp. 29–42, 2012.

[46] A. Pathak, Y. C. Hu, M. Zhang, P. Bahl, and Y.-M. Wang, "Fine-grained power modeling for smartphones using system call tracing," pp. 153–168, 2011.

[47] F. Qian, Z. Wang, A. Gerber, Z. Mao, S. Sen, and O. Spatscheck, "Profiling resource usage for mobile applications: a cross-layer approach," pp. 321–334, 2011.

[48] A. Razaghpanah, R. Nithyanand, N. Vallina-Rodriguez, S. Sundaresan, M. Allman, C. Kreibich, and P. Gill, "Apps, trackers, privacy, and regulators: A global study of the mobile tracking ecosystem," 2018.

[49] M. Elsabagh, R. Johnson, A. Stavrou, C. Zuo, Q. Zhao, and Z. Lin, "{FIRMSCOPE}: Automatic uncovering of privilege-escalation vulnerabilities in pre-installed apps in android firmware," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020.

[50] J. Ren, A. Rao, M. Lindorfer, A. Legout, and D. Choffnes, "Recon: Revealing and controlling pii leaks in mobile network traffic," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, pp. 361–374, 2016.

[51] L. Verderame, D. Caputo, A. Romdhana, and A. Merlo, "On the (un) reliability of privacy policies in android apps," *arXiv preprint arXiv:2004.08559*, 2020.

[52] M. Lécuyer, G. Ducoffe, F. Lan, A. Papancea, T. Petsios, R. Spahn, A. Chaintreau, and R. Geambasu, "Xray: Enhancing the web's transparency with differential correlation," 2014.

[53] M. Gandhi, M. Jakobsson, and J. Ratkiewicz, "Badvertisements: Stealthy click-fraud with unwitting accessories," *Journal of Digital Forensic Practice*, vol. 1, no. 2, pp. 131–142, 2006.

[54] S. Guha, B. Cheng, and P. Francis, "Challenges in measuring online advertising systems," pp. 81–87, 2010.

[55] A. Datta, M. C. Tschantz, and A. Datta, "Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination," *arXiv preprint arXiv:1408.6491*, 2014.

[56] A. Rao, F. Schaub, and N. Sadeh, "What do they know about me? contents and concerns of online behavioral profiles," *arXiv preprint arXiv:1506.01675*, 2015.

[57] T. Book and D. S. Wallach, "An empirical study of mobile ad targeting," *arXiv preprint arXiv:1502.06577*, 2015.

[58] R. Stevens, C. Gibler, J. Crussell, J. Erickson, and H. Chen, "Investigating user privacy in android ad libraries," 2012.

[59] X. Liu, J. Liu, S. Zhu, W. Wang, and X. Zhang, "Privacy risk analysis and mitigation of analytics libraries in the android ecosystem," *IEEE Transactions on Mobile Computing*, 2019.

[60] P. Pearce, A. P. Felt, G. Nunez, and D. Wagner, "Addroid: Privilege separation for applications and advertisers in android," pp. 71–72, 2012.

[61] S. Shekhar, M. Dietz, and D. S. Wallach, "Adsplit: Separating smartphone advertising from applications.," pp. 553–567, 2012.

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

[63] G. Aggarwal, S. Muthukrishnan, D. Pál, and M. Pál, "General auction mechanism for search advertising," pp. 241–250, 2009.

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

[65] B. Krishnamurthy and C. E. Wills, "On the leakage of personally identifiable information via online social networks," pp. 7–12, 2009.

[66] B. Krishnamurthy and C. E. Wills, "Privacy leakage in mobile online social networks," pp. 4–4, 2010.

[67] A. Metwally, D. Agrawal, and A. El Abbadi, "Detectives: detecting coalition hit inflation attacks in advertising networks streams," pp. 241–250, 2007.

[68] Y. Wang, D. Burgener, A. Kuzmanovic, and G. Maciá-Fernández, "Understanding the network and user-targeting properties of web advertising networks," pp. 613–622, 2011.

[69] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, M. E. Seligman, *et al.*, "Personality, gender, and age in the language of social media: The open-vocabulary approach," *PloS one*, vol. 8, no. 9, p. e73791, 2013.

[70] M. Kosinski, D. Stillwell, and T. Graepel, "Private traits and attributes are predictable from digital records of human behavior," *Proceedings of the National Academy of Sciences*, vol. 110, no. 15, pp. 5802–5805, 2013.

[71] S. Goel, J. M. Hofman, and M. I. Sirer, "Who does what on the web: A large-scale study of browsing behavior.," in *ICWSM*, 2012.

[72] J. Hu, H.-J. Zeng, H. Li, C. Niu, and Z. Chen, "Demographic prediction based on user's browsing behavior," in *Proceedings of the 16th international conference on World Wide Web*, pp. 151–160, ACM, 2007.

[73] J. Schler, M. Koppel, S. Argamon, and J. W. Pennebaker, "Effects of age and gender on blogging.," in *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, vol. 6, pp. 199–205, 2006.

[74] J. Otterbacher, "Inferring gender of movie reviewers: exploiting writing style, content and metadata," in *Proceedings of the 19th ACM international conference on Information and knowledge management*, pp. 369–378, ACM, 2010.

[75] A. Mukherjee and B. Liu, "Improving gender classification of blog authors," in *Proceedings of the 2010 conference on Empirical Methods in natural Language Processing*, pp. 207–217, Association for Computational Linguistics, 2010.

[76] B. Bi, M. Shokouhi, M. Kosinski, and T. Graepel, "Inferring the demographics of search users: social data meets search queries," in *Proceedings of the 22nd international conference on World Wide Web*, pp. 131–140, International World Wide Web Conferences Steering Committee, 2013.

[77] J. J.-C. Ying, Y.-J. Chang, C.-M. Huang, and V. S. Tseng, "Demographic prediction based on users mobile behaviors," *Mobile Data Challenge*, 2012.

[78] J. W. Pennebaker, M. E. Francis, and R. J. Booth, "Linguistic inquiry and word count: Liwc 2001," *Mahway: Lawrence Erlbaum Associates*, vol. 71, p. 2001, 2001.

[79] E. Zheleva and L. Getoor, "To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles," pp. 531–540, 2009.

[80] J. He, W. W. Chu, and Z. V. Liu, "Inferring privacy information from social networks," pp. 154–165, 2006.

[81] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: inferring user profiles in online social networks," pp. 251–260, 2010.

[82] E. Ryu, Y. Rong, J. Li, and A. Machanavajjhala, "curso: protect yourself from curse of attribute inference: a social network privacy-analyzer," in *Proceedings of the ACM SIGMOD Workshop on Databases and Social Networks*, pp. 13–18, ACM, 2013.

[83] P. Samarati, "Protecting respondents identities in microdata release," *IEEE transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2001.

[84] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[85] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, p. 3, 2007.

[86] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," pp. 106–115, 2007.

[87] C. Aguilar Melchor and P. Gaborit, "A lattice based computationally efficient private information retrieval protocol," vol. 446, 2007.

[88] B. Chor and N. Gilboa, "Computationally private information retrieval," pp. 304–313, 1997.

[89] I. Goldberg, "Improving the robustness of private information retrieval," pp. 131–148, 2007.

[90] R. Henry, F. Olumofin, and I. Goldberg, "Practical pir for electronic commerce," pp. 677–690, 2011.

[91] A. Beimel, Y. Ishai, and T. Malkin, "Reducing the servers computation in private information retrieval: Pir with preprocessing," *Journal of Cryptology*, vol. 17, no. 2, pp. 125–151, 2004.

[92] Y. Gertner, S. Goldwasser, and T. Malkin, "A random server model for private information retrieval," pp. 200–217, 1998.

[93] C. Devet, I. Goldberg, and N. Heninger, "Optimally robust private information retrieval.," pp. 269–283, 2012.

[94] C. Devet and I. Goldberg, "The best of both worlds: Combining information-theoretic and computational pir for communication efficiency," pp. 63–82, 2014.

[95] W. Enck, P. Gilbert, S. Han, V. Tendulkar, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth, "Taintdroid: an information-flow tracking system for realtime privacy monitoring on smartphones," *ACM Transactions on Computer Systems (TOCS)*, vol. 32, no. 2, p. 5, 2014.

[96] M. Ongtang, S. McLaughlin, W. Enck, and P. McDaniel, "Semantically rich application-centric security in android," *Security and Communication Networks*, vol. 5, no. 6, pp. 658–673, 2012.

[97] A. Frik, A. Haviland, and A. Acquisti, "The impact of adblockers on product search and purchase behavior: A lab experiment," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020.

[98] A. Shuba and A. Markopoulou, "Nomoats: Towards automatic detection of mobile tracking," *Proceedings on Privacy Enhancing Technologies*, vol. 2, pp. 45–66, 2020.

[99] U. Iqbal, P. Snyder, S. Zhu, B. Livshits, Z. Qian, and Z. Shafiq, "Adgraph: A graph-based approach to ad and tracker blocking," in *Proc. of IEEE Symposium on Security and Privacy*, 2020.

[100] A. P. Felt, E. Ha, S. Egelman, A. Haney, E. Chin, and D. Wagner, "Android permissions: User attention, comprehension, and behavior," in *Proceedings of the eighth symposium on usable privacy and security*, pp. 1–14, 2012.

[101] A. P. Felt, H. J. Wang, A. Moshchuk, S. Hanna, and E. Chin, "Permission Re-Delegation: Attacks and Defenses," 2011.

[102] A. P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner, "Android permissions demystified," in *Proceedings of the 18th ACM conference on Computer and communications security*, pp. 627–638, 2011.

[103] P. P. Chan, L. C. Hui, and S.-M. Yiu, "Droidchecker: analyzing android applications for capability leak," in *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks*, pp. 125–136, 2012.

[104] W. Enck, M. Ongtang, and P. McDaniel, "On lightweight mobile phone application certification," in *Proceedings of the 16th ACM conference on Computer and communications security*, pp. 235–245, 2009.

[105] A. R. Beresford, A. Rice, N. Skehin, and R. Sohan, "Mockdroid: trading privacy for application functionality on smartphones," pp. 49–54, 2011.

[106] 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 *Proceedings of the 18th ACM conference on Computer and communications security*, pp. 639–652, 2011.

[107] P. Golle and K. Partridge, "On the anonymity of home/work location pairs," in *International Conference on Pervasive Computing*, pp. 390–397, Springer, 2009.

[108] H. Zang and J. Bolot, "Anonymization of location data does not work: A large-scale measurement study," in *Proceedings of the 17th annual international conference on Mobile computing and networking*, pp. 145–156, 2011.

[109] N. Mohammed, B. C. Fung, and M. Debbabi, "Walking in the crowd: anonymizing trajectory data for pattern analysis," in *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 1441–1444, 2009.

[110] F. Bonchi, L. V. Lakshmanan, and H. Wang, "Trajectory anonymity in publishing personal mobility data," *ACM Sigkdd Explorations Newsletter*, vol. 13, no. 1, pp. 30–42, 2011.

[111] R. Shokri, G. Theodorakopoulos, G. Danezis, J.-P. Hubaux, and J.-Y. Le Boudec, "Quantifying location privacy: the case of sporadic location exposure," in *International Symposium on Privacy Enhancing Technologies Symposium*, pp. 57–76, Springer, 2011.

[112] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography conference*, pp. 265–284, Springer, 2006.

[113] C. Dwork, A. Roth, *et al.*, "The algorithmic foundations of differential privacy.," *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211–407, 2014.

[114] H. Cho, D. Ippolito, and Y. W. Yu, "Contact tracing mobile apps for covid-19: Privacy considerations and related trade-offs," *arXiv preprint arXiv:2003.11511*, 2020.

[115] Y. Yan, X. Gao, A. Mahmood, T. Feng, and P. Xie, "Differential private spatial decomposition and location publishing based on unbalanced quadtree partition algorithm," *IEEE Access*, vol. 8, pp. 104775–104787, 2020.

[116] X. Zhang, R. Chen, J. Xu, X. Meng, and Y. Xie, "Towards accurate histogram publication under differential privacy," in *Proceedings of the 2014 SIAM international conference on data mining*, pp. 587–595, SIAM, 2014.

[117] J. Zhang, X. Xiao, and X. Xie, "Privtree: A differentially private algorithm for hierarchical decompositions," in *Proceedings of the 2016 International Conference on Management of Data*, pp. 155–170, 2016.

[118] E. Kushilevitz and R. Ostrovsky, "Replication is not needed: Single database, computationally-private information retrieval," pp. 364–364, 1997.

[119] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, "Private information retrieval," *http://dl.acm.org/citation.cfm?id=795662.796270*, pp. 41–, 1995.

[120] B. Chor, N. Gilboa, and M. Naor, "Private information retrieval by keywords," 1997.

[121] C.-K. Chu and W.-G. Tzeng, "Efficient k-out-of-n oblivious transfer schemes.," *J. UCS*, vol. 14, no. 3, pp. 397–415, 2008.

[122] M. Naor and B. Pinkas, "Oblivious transfer and polynomial evaluation," pp. 245–254, 1999.

[123] H. Mozaffari and A. Houmansadr, "Heterogeneous private information retrieval,"

[124] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[125] V. Guruswami and A. Rudra, "Explicit codes achieving list decoding capacity: Error-correction with optimal redundancy," *Information Theory, IEEE Transactions on*, vol. 54, no. 1, pp. 135–150, 2008.

[126] P. Mittal, F. G. Olumofin, C. Troncoso, N. Borisov, and I. Goldberg, "Pir-tor: Scalable anonymous communication using private information retrieval.," 2011.

[127] A. Beimel and Y. Stahl, "Robust information-theoretic private information retrieval," pp. 326–341, 2003.

[128] A. Beimel and Y. Stahl, "Robust information-theoretic private information retrieval," *Journal of Cryptology*, vol. 20, no. 3, pp. 295–321, 2007.

[129] S. Micali, C. Peikert, M. Sudan, and D. A. Wilson, "Optimal error correction against computationally bounded noise," pp. 1–16, 2005.

[130] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," pp. 514–532, 2001.

[131] A. Kate, G. M. Zaverucha, and I. Goldberg, "Constant-size commitments to polynomials and their applications," pp. 177–194, 2010.

[132] A. Kate, G. M. Zaverucha, and I. Goldberg, "Polynomial commitments," 2010.

[133] F. Boudot, "Efficient proofs that a committed number lies in an interval," pp. 431–444, 2000.

[134] C.-P. Schnorr, "Efficient identification and signatures for smart cards," pp. 239–252, 1990.

[135] S. A. Brands, "Rethinking public key infrastructures and digital certificates: building in privacy," 2000.

[136] J. Camenisch and M. Michels, "Proving in zero-knowledge that a number is the product of two safe primes," pp. 107–122, 1999.

[137] M. Bellare, J. A. Garay, and T. Rabin, "Fast batch verification for modular exponentiation and digital signatures," in *Advances in Cryptology-EUROCRYPT'98*, pp. 236–250, Springer, 1998.

[138] M. Bellare, J. A. Garay, and T. Rabin, "Batch verification with applications to cryptography and checking," pp. 170–191, 1998.

[139] M. Fredrikson and B. Livshits, "Repriv: Re-imagining content personalization and in-browser privacy," pp. 131–146, 2011.

[140] S. Guha, B. Cheng, and P. Francis, "Privad: Practical privacy in online advertising.," 2011.

[141] R. Chen, A. Reznichenko, P. Francis, and J. Gehrke, "Towards statistical queries over distributed private user data," in *Presented as part of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, pp. 169–182, 2012.

[142] R. Chen, I. E. Akkus, and P. Francis, "Splitx: high-performance private analytics," pp. 315–326, 2013.

[143] M. M. Tsang, S.-C. Ho, and T.-P. Liang, "Consumer attitudes toward mobile advertising: An empirical study," *International journal of electronic commerce*, vol. 8, no. 3, pp. 65–78, 2004.

[144] M. Merisavo, S. Kajalo, H. Karjaluoto, V. Virtanen, S. Salmenkivi, M. Raulas, and M. Leppäniemi, "An empirical study of the drivers of consumer acceptance of mobile advertising," *Journal of interactive advertising*, vol. 7, no. 2, pp. 41–50, 2007.

[145] G. A. Johnson, S. K. Shriver, and S. Du, "Consumer privacy choice in online advertising: Who opts out and at what cost to industry?," *Marketing Science*, 2020.

[146] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," 2004.

[147] G. Aggarwal, E. Bursztein, C. Jackson, and D. Boneh, "An analysis of private browsing modes in modern browsers.," pp. 79–94, 2010.

[148] I. E. Akkus, R. Chen, M. Hardt, P. Francis, and J. Gehrke, "Non-tracking web analytics," 2012.

[149] M. Backes, A. Kate, M. Maffei, and K. Pecina, "Obliviad: Provably secure and practical online behavioral advertising," pp. 257–271, 2012.

[150] M. Hardt and S. Nath, "Privacy-aware personalization for mobile advertising," 2012.

[151] P. Samarati and L. Sweeney, "Generalizing data to provide anonymity when disclosing information," in *PODS*, vol. 98, p. 188, 1998.

[152] S. R. Ganta, S. P. Kasiviswanathan, and A. Smith, "Composition attacks and auxiliary information in data privacy," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 265–273, ACM, 2008.

[153] L. Sweeney, "Simple demographics often identify people uniquely," *Health (San Francisco)*, vol. 671, pp. 1–34, 2000.

[154] S. E. Coull, C. V. Wright, F. Monrose, M. P. Collins, M. K. Reiter, *et al.*, "Playing devil's advocate: Inferring sensitive information from anonymized network traces.," in *NDSS*, vol. 7, pp. 35–47, 2007.

[155] H. Artail and R. Farhat, "A privacy-preserving framework for managing mobile ad requests and billing information," *Mobile Computing, IEEE Transactions on*, vol. 14, no. 8, pp. 1560–1572, 2015.

[156] M. Hardt and S. Nath, "Privacy-aware personalization for mobile advertising," in *Proceedings of the 2012 ACM conference on Computer and communications security*, pp. 662–673, ACM, 2012.

[157] D. Wermke, N. Huaman, Y. Acar, B. Reaves, P. Traynor, and S. Fahl, "A large scale investigation of obfuscation use in google play," in *Proceedings of the 34th Annual Computer Security Applications Conference*, pp. 222–235, 2018.

[158] U. Weinsberg, S. Bhagat, S. Ioannidis, and N. Taft, "Blurme: inferring and obfuscating user gender based on ratings," pp. 195–202, 2012.

[159] S. Salamatian, A. Zhang, F. du Pin Calmon, S. Bhamidipati, N. Fawaz, B. Kveton, P. Oliveira, and N. Taft, "How to hide the elephant-or the donkey-in the room: Practical privacy against statistical inference for large data," *IEEE GlobalSIP*, 2013.

[160] F. du Pin Calmon and N. Fawaz, "Privacy against statistical inference," pp. 1401–1408, 2012.

[161] C. Li, H. Shirani-Mehr, and X. Yang, "Protecting individual information against inference attacks in data publishing," pp. 422–433, 2007.

[162] D. C. Howe and H. Nissenbaum, "Trackmenot: Resisting surveillance in web search," *Lessons from the Identity Trail: Anonymity, Privacy, and Identity in a Networked Society*, vol. 23, pp. 417–436, 2009.

[163] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *ACM Sigmod Record*, vol. 29, pp. 439–450, ACM, 2000.

[164] A. Evfimievski, J. Gehrke, and R. Srikant, "Limiting privacy breaches in privacy preserving data mining," in *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pp. 211–222, ACM, 2003.

[165] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, "On the privacy preserving properties of random data perturbation techniques," in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pp. 99–106, IEEE, 2003.

[166] N. Mor, O. Riva, S. Nath, and J. Kubiatowicz, "Bloom cookies: Web search personalization without user tracking.," in *NDSS*, 2015.

[167] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM*, vol. 13, no. 7, pp. 422–426, 1970.
[168] C. Dwork, "Differential privacy," in *Automata, languages and programming*, pp. 1–12, Springer, 2006.
[169] T. Dierks, "The transport layer security (tls) protocol version 1.2," 2008.
[170] V. Rastogi and S. Nath, "Differentially private aggregation of distributed time-series with transformation and encryption," pp. 735–746, 2010.
[171] E. Shi, T. H. Chan, E. Rieffel, R. Chow, and D. Song, "Privacy-preserving aggregation of time-series data," in *Proc. NDSS*, vol. 2, pp. 1–17, 2011.
[172] X. Yi, R. Paulet, and E. Bertino, *Homomorphic Encryption and Applications*. Springer, 2014.
[173] J. Ghaderi and R. Srikant, "Towards a theory of anonymous networking," in *INFOCOM, 2010 Proceedings IEEE*, pp. 1–9, IEEE, 2010.
[174] A. Juels, "Targeted advertising... and privacy too," in *Topics in Cryptology CT-RSA 2001*, pp. 408–424, Springer, 2001.
[175] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–90, 1981.
[176] Y. Desmedt and K. Kurosawa, "How to break a practical mix and design a new one," in *Advances in Cryptology EUROCRYPT 2000*, pp. 557–572, Springer, 2000.
[177] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pp. 218–229, ACM, 1987.
[178] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," tech. rep., Manubot, 2019.
[179] V. Dedeoglu, R. Jurdak, A. Dorri, R. Lunardi, R. Michelin, A. Zorzo, and S. Kanhere, "Blockchain technologies for iot," in *Advanced Applications of Blockchain Technology*, pp. 55–89, Springer, 2020.
[180] J. Yang, J. Wen, B. Jiang, and H. Wang, "Blockchain-based sharing and tamper-proof framework of big data networking," *IEEE Network*, vol. 34, no. 4, pp. 62–67, 2020.
[181] A. Tandon, A. Dhir, N. Islam, and M. Mäntymäki, "Blockchain in healthcare: A systematic literature review, synthesizing framework and future research agenda," *Computers in Industry*, vol. 122, p. 103290, 2020.
[182] Y. Chen and C. Bellavitis, "Blockchain disruption and decentralized finance: The rise of decentralized business models," *Journal of Business Venturing Insights*, vol. 13, p. e00151, 2020.
[183] J. Freudiger, N. Vratonjic, and J.-P. Hubaux, "Towards privacy-friendly online advertising," no. LCA-CONF-2009-008, 2009.
[184] I. E. Akkus, R. Chen, M. Hardt, P. Francis, and J. Gehrke, "Non-tracking web analytics," pp. 687–698, 2012.
[185] S. Christopher, S. Sid, and K. Dan
[186] A. Ghosh and A. Roth, "Selling privacy at auction," *Games and Economic Behavior*, 2013.
[187] C. Riederer, V. Erramilli, A. Chaintreau, B. Krishnamurthy, and P. Rodriguez, "For sale: your data: by: you," in *Proceedings of the 10th ACM WORKSHOP on Hot Topics in Networks*, p. 13, ACM, 2011.
[188] M. A. Bashir, S. Arshad, W. Robertson, and C. Wilson, "Tracing information flows between ad exchanges using retargeted ads," in *25th {USENIX} Security Symposium ({USENIX} Security 16)*, pp. 481–496, 2016.
[189] W. Melicher, M. Sharif, J. Tan, L. Bauer, M. Christodorescu, and P. G. Leon, "(do not) track me sometimes: Users' contextual preferences for web tracking," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 2, pp. 135–154, 2016.

Imdad Ullah (Member, IEEE) has received his Ph.D. in Computer Science and Engineering from The University of New South Wales (UNSW) Sydney, Australia. He is currently an assistant professor with the College of Computer Engineering and Sciences, PSAU, Saudi Arabia. He has served in various positions of Researcher at UNSW, Research scholar at National ICT Australia (NICTA)/Data61 CSIRO Australia, NUST Islamabad Pakistan and SEEMOO TU Darmstadt Germany, and Research Collaborator at SLAC National Accelerator Laboratory Stanford University USA. He has research and development experience in privacy preserving systems including private advertising and crypto-based billing systems. His primary research interest include privacy enhancing technologies; he also has interest in Internet of Things, Blockchain, network modeling and design, network measurements, and trusted networking.

Roksana Boreli has received her Ph.D in Communications from University of Technology, Sydney, Australia. She has over 20 years of experience in communications and networking research and in engineering development, in large telecommunications companies (Telstra Australia, Xantic, NL) and research organisations. Roksana has served in various positions of Engineering manager, Technology strategist, Research leader of the Privacy area of Networks research group in National ICT Australia (NICTA)/CSIRO Data61 and CTO in a NICTA spinoff 7-ip. Her primary research focus is on the privacy enhancing technologies; she also maintains an interest in mobile and wireless communications.

Salil S. Kanhere (Senior Member, IEEE) received the M.S. and Ph.D. degrees from Drexel University, Philadelphia. He is currently a Professor of Computer Science and Engineering with UNSW Sydney, Australia. His research interests include the Internet of Things, cyberphysical systems, blockchain, pervasive computing, cybersecurity, and applied machine learning. He is a Senior Member of the ACM, an Humboldt Research Fellow, and an ACM Distinguished Speaker. He serves as the Editor in Chief of the Ad Hoc Networks journal and as an Associate Editor of the IEEE Transactions On Network and Service Management, Computer Communications, and Pervasive andMobile Computing. He has served on the organising committee of several IEEE/ACM international conferences. He has co-authored a book titled *Blockchain for Cyberphysical Systems*.