Published at the 16th ACM Workshop on Privacy in the Electronic Society (WPES 2017). More information at https://adint.cs.washington.edu/.

Exploring ADINT: Using Ad Targeting for Surveillance on a Budget — or — How Alice Can Buy Ads to Track Bob

Paul Vines, Franziska Roesner, and Tadayoshi Kohno
Paul G. Allen School of Computer Science & Engineering, University of Washington
plvines@cs.washington.edu,franzi@cs.washington.edu,yoshi@cs.washington.edu

ABSTRACT

The online advertising ecosystem is built upon the ability of advertising networks to know properties about users (e.g., their interests or physical locations) and deliver targeted ads based on those properties. Much of the privacy debate around online advertising has focused on the harvesting of these properties by the advertising networks. In this work, we explore the following question: can third-parties use the *purchasing* of ads to extract private information about individuals? We find that the answer is yes. For example, in a case study with an archetypal advertising network, we find that — for $1000 USD — we can track the location of individuals who are using apps served by that advertising network, as well as infer whether they are using potentially sensitive applications (e.g., certain religious or sexuality-related apps). We also conduct a broad survey of other ad networks and assess their risks to similar attacks. We then step back and explore the implications of our findings.

CCS CONCEPTS

• **Security and privacy → Social aspects of security and privacy**; *Privacy protections*; Mobile and wireless security;

KEYWORDS

ADINT, Location, Online Advertising, Privacy, Surveillance, Targeted Advertising

1 INTRODUCTION

Much of the debate around online advertising has focused on the collection of private information about users by advertising networks, and on the use of that information for targeted advertising. However, there exist other threats — threats in which *regular people*, not just impersonal, commercially-motivated merchants or advertising networks — can exploit the online advertising ecosystem to extract private information about *other people*, such as people that they know or that live nearby. We explore this threat in this paper.

Our study has three key elements:

Element 1: Surfacing Advertising-based Information Collection as a Threat. We identify and discuss the privacy threats posed by third-party *purchasers* of ads, which could be regular individuals and not large, impersonal corporations. By surfacing and diving deeply into this threat, we hope that our work can contribute a fresh perspective to the web tracking and online advertising debate.

For expository convenience, we sought an acronym to refer to our method of extracting information about targeted individuals through the purchasing of ads. We observe that governments use the word "intelligence" to refer to the collection of information about targets, and have a rich set of acronyms for different methods of intelligence collection, e.g., SIGINT and HUMINT [7]. Inspired by this terminology, we use the term *ADINT* to refer to our method of exploiting the advertising ecosystem, as the purchaser of ads, to collect information about targeted individuals.

Element 2: Evaluating ADINT Capabilities. We conduct a deep case study to gauge actual ADINT capabilities, using a canonical demand-side provider (DSP, the entities that provide targeted advertising) in Section 4. To complement this deep dive, we perform an analysis of 20 other DSPs, to identify and explore the breadth of ADINT capabilities available (Section 5).

Element 3: Study of Implications. We step back and explore the implications of our findings, and of ADINT in general, to key stakeholders, including people who might use ADINT, potential victims, advertisers, and policy makers (Section 6).

Example Results. To foreshadow some results, we find that an individual or small group with a $1000 US Dollar budget can use targeted ads and a DSP to track the locations of targeted individuals as they move from home, to work, and to other sensitive locations. We find that we can target ads to users of specific applications and at specific locations, which means that one can use purchased ads to count the number of Grindr (a gay online dating app) users or Quran Reciters (a religious app) users in a house. We find that we can use targeted ads to learn when a person is using a specific application (e.g., when a targeted individual is using Talkatone, a messaging app); a natural extension could be to observe whether two targeted individuals are using the same app at the same time, thereby yielding potential side-channel information about communications patterns. Building on our broad analysis of 20 other DSPs, we further identify numerous other information-extraction capabilities.

2 BACKGROUND & RELATED WORK

Online Advertising Background. Getting an ad to a user on a webpage or app today is a complex task involving a number of distinct entities (Figure 1).

Audience and Publisher. The audience is the users that will see the ads. They see ads when they interact with content from a publisher. Publishers are the owners of websites or apps. The publisher

WPES'17, October 30, 2017, Dallas, TX, USA

ACM ISBN 978-1-4503-5175-1/17/10…$15.00
https://doi.org/10.1145/3139550.3139567

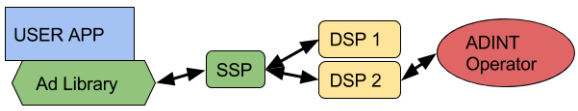


Figure 1: Overview of the ad-serving process including an ADINT attacker as the advertiser. Arrows are HTTP(S) requests and responses.

generally includes ad-libraries (provided by SSPs, below) in their website or app to serve ads [11].

Supply-Side Provider (SSP). The supply-side provider (SSP) manages publishers and facilitates selling ad inventory — the ad-spaces in a publisher's content — by auctioning it to Demand-Side Providers (DSPs, below) [5]. Ads are more effective at "converting" an audience (e.g., buying something) when they are specifically targeted. Thus, the more information an SSP can provide to bidders the higher the bids they will get. SSPs often provide the ad-libraries that publishers put in their content to perform this information-gathering and ad inventory auctioning automatically [24]. The most basic tracking information these ad-libraries send the SSP is a cookie or Mobile Advertising ID (MAID), described below.

Demand-Side Provider (DSP). The demand-side provider (DSP) manages advertisers and bids on ad inventory from SSPs it is connected to [5]. Advertisers are entities that have advertisements they want shown. DSPs facilitate purchasing an ad slot and serving an ad on behalf of an advertiser. Depending on the DSP, the advertiser will upload the actual ad content they want shown or they can host it on their own servers and provide a URL for the DSP.

The DSP also provides *ad targeting* on behalf of an advertiser. The types of targeting DSPs provide can vary greatly; we evaluate targeting options across different DSPs in Section 5. The information used for the targeting decision can come from several places: (1) the information the SSP gathers directly from the audience may be forwarded to the DSPs; (2) the DSPs may keep their own data about individuals and reidentify the audience based on information provided by the SSP (such as cookies or a MAID, see below); (3) the DSP could also use a data management platform (DMP) to provide information about an audience.

Cookies & Mobile Advertising ID (MAID). Third-party cookies are the classic method for tracking audiences [3]. Typically an advertiser facilitates a DSP setting a cookie on a user when they visit the advertiser's website.

For ADINT purposes we typically do not expect targets (a.k.a., victims) to visit the ADINT attacker's website. Nevertheless, active types of ad content, such as Flash or HTML5 ads, can make requests independent of target interaction which could set cookies.

The Mobile Advertising ID (MAID) has a purpose similar to tracking cookies. Because of the architecture of mobile operating systems, each app has its own cookie store. This makes each app appear as a different user to traditional cookie tracking and hinders targeted advertising. The solution to this is a sort of whole-device cookie. Originally this relied on permanent device identifiers that were not practically user-resettable. However, on Android phones that use the Google PlayStore, the Google Advertising ID (GAID) has now been introduced that provides an ability to reset the identifier from deep within the settings app.

Related Works. The advertising ecosystem and its security and privacy implications has been a significant area of research, e.g., [1, 2, 12, 15, 21, 26]. Malicious use of advertising *content* (malvertising) has recently been on the rise [16, 18, 28, 29]. Additionally, one recent work showed that some ad-libraries allow sensitive data to be extracted without even exploiting them [22]. The normal behavior of mobile ad-libraries has also been studied [9, 22] and these studies show that ad-libraries often have poor security as well as fraudulent behavior which could allow advanced ADINT attacks additional capabilities (see Section 5).

Englehardt et al. examined the capabilities of an intelligence agency, in a privileged but passive network position, to track user browsing via intercepting the tracking cookies used by advertising networks [13]. In 2010 Korolova conducted an advertising inference attack to learn information users had uploaded to Facebook but not shared publicly. Their attack showed the idea of extracting information via ad targeting was feasible. Since 2010, Facebook and the rest of the advertising ecosystem have changed significantly; following the Snowden disclosures many technology companies sought to portray a more user-friendly and anti-surveillance image [17, 19], but it is unclear if this has impacted ad tracking. Additionally, Korolova showed an attack on Facebook users; however, as both a social media platform and ad-network, Facebook is an anomaly in the advertising ecosystem and so attacks against it do not necessarily generalize to most ad-networks. For example, Korolova extracted information that users *explicitly* chose to share with some entity — information added to their Facebook profiles — albeit not necessarily publicly or with the attacker. We focus on more typical ad-networks, which could not have been used to perform Korolova's attack because of different targeting capabilities.

3 RESEARCH QUESTIONS

We define ADINT as the use of the online advertising ecosystem to collect sensitive information about targets (victims), where the attacker collecting that information is doing so by purchasing ads. This work is therefore about scientifically studying, and evaluating, the capabilities of ADINT. In particular, we sought answers to the following questions:

(1) *Possibility.* Is ADINT even possible? Can an attacker purchase online ads from a DSP and, as a result of the standard DSP ad display and reporting process, harvest intelligence about targeted individuals (a.k.a., the target, the victim)?
(2) *Capabilities.* What types of information can the attacker obtain about targeted individuals using ADINT?
(3) *Operational Aspects.* If ADINT is possible, then what are the resources required for a successful ADINT campaign, including cost and any necessary preconditions for the attack to be successful? How reliable is ADINT? How efficient?

To foreshadow answers to these questions, we find that, for our example DSP, an attacker first needs to learn the device identifier for a target's mobile phone. The attacker can learn the target's mobile phone device identifier in a number of ways, as we explore. Subsequently, after a $1000 deposit, the attacker can learn if a target visits a pre-defined sensitive location within 10 minutes of the target's arrival at that location with high reliability if the target briefly uses an applicable mobile phone application at that location.

In the above description, location determination is an example of a *capability*, and the need to first learn the device identifier of the phone, the cost, and the time required are example *operation aspects*.

We expand on these goals in subsequent sections, as we explore and define the ADINT threat model in more depth. We explore these questions in two ways. First, we conduct an in-depth case study of a single DSP. This case study enables us to empirically evaluate the constraints of an archetypal advertising platform. Second, we survey 20 other DSPs, with the goal of developing a rich and broad perspective on the full extent of ADINT capabilities.

4 CASE STUDY: AN ARCHETYPICAL DSP

To answer our research questions about the real world capabilities of ADINT, we conducted a case study of an archetypical DSP. This DSP has a moderate cost, and supports a diversity of targeting criteria, including targeting users based on location and demographics. Table 5 in Section 5 provides an overview comparison of this DSP to other DSPs. We do not name our case study DSP because, as we discuss in Section 5, our paper's overall results are industry-wide and not restricted to this DSP. We focus our analysis on an attacker's ability to infer the location of a target because, unlike the target's demographics or what apps the target chooses to use, location is highly dynamic. We turn to these other types of information inference toward the end of this section.

We conducted our evaluation in two stages. First, we devised a set of benchmarking experiments to develop an understanding of the capabilities and limits of our DSP under controlled conditions. Second, we created a set of realistic, end-to-end proof-of-concept attack scenarios based on capabilities determined in our benchmarks, and we used these scenarios to evaluate more concrete ADINT attacks.

4.1 Case Study Threat Model

We now define the threat model for our case study. This threat model is somewhat abstract but most closely resembles a stalker. Potential uses of ADINT by other types of real attackers are discussed in Section 6.

The attacker's goal is to remotely surveil a specific target over time and obtain sensitive information about that target. The attacker wants to know where the target goes, where they live, and other sensitive information such as what apps they use (which could reveal information about them as people).

In this threat model the ADINT attacker requires several preconditions to be true:

(1) The attacker can serve ads.
(2) The target uses a smartphone or other mobile device and ad-containing apps that our DSP can serve to.
(3) The attacker knows the target's device's Mobile Advertising ID (MAID) for some attacks.

For precondition (1), to serve ads with our DSP the attacker needs $1000 for a deposit and to possess a website for the ads to direct to.

For precondition (2), our DSP, like many DSPs, serves ads to numerous popular apps. Table 1 summarizes the apps that we tested. We explore ADINT's use in the desktop environment and in web ads in Section 5 and 6 but focus on mobile ads here. Mobile ads are particularly interesting for location attacks, since people move with their devices.

For precondition (3), obtaining the target's MAID can be done in several ways. In our experiments, (A) we sniff network traffic of target devices to obtain the MAID, which is often sent to ad-exchanges unencrypted. Examples of an attacker that could do this include: anyone temporarily in WiFi range of the target when they are on an unsecured network; similarly anyone capable of temporarily intercepting cellular traffic of the target (an increasingly easy attack [6, 10]); or anyone with temporary access to the WiFi router the target uses. An important aspect of all three of these scenarios is that the attacker only needs to perform this step once and can then perform ADINT attacks on the target at arbitrary distances and while they are connected to arbitrary networks. Additionally, (B) we experimentally verified that an attacker can also obtain the MAID if the target clicks on any of the attacker's earlier ads. The MAID can (C) also be exfiltrated via JavaScript in ads in some major ad-libraries [22]. Although we did not do so, it is also possible to (D) purchase the target's MAID online [14]. Further, as we will discuss later, precondition (3) is not necessary for certain attacks.

Our threat model does *not* assume the target will interact with the ads in any manner. Furthermore, we do not include any active content — such as JavaScript or Flash code — in our ads. Refraining from active content allows our case study methods to apply to other DSPs, even those that only allow static image ad content, as some do (see Section 5 for more details). Only using static image ads also shows our attacks are not dependent on client-side details such as which ad-library our ad is served to.

The targeting-based ADINT attackers that we evaluate here are composable with active ad content to create enhanced ADINT capabilities; we explore this extension of ADINT in Sections 5.

4.2 Methodology

We used a mix of 10 facsimile user devices and 10 real user devices in our evaluation. The former enabled rigorous testing of inconvenient scenarios, the latter enabled our study to reflect in-the-wild results. We created our facsimile users as new user accounts of 27 year-old females on factory-reset Moto-G smartphones running Android 4.4.4 with new SIM cards. We connected the devices to local WiFi networks and downloaded the apps we evaluate, as well as apps for capturing ground truth GPS and network data from the device. Finally, we gathered the Mobile Advertising ID (MAID) for each facsimile device to use for ad targeting. We also used Android phones of real users; we ensured that the phones' owners understood what we planned to do and we took precautions to avoid learning any personal information about the phones as part of our study (e.g., we did not record what ads were displayed except our own, and we reset the user MAID after the study). We evaluated these phones on benchmarks 2, 3, and 4 (see below) to determine the cost and frequency we could target real users, in the wild, with ads.

Apps Tested. We selected apps to test by analyzing a list of apps our DSP could serve ads to. Since we were primarily focused on location-targeting, we selected an app to conduct the majority of our evaluations on that had the largest user-base and also allowed location targeting. This app was Talkatone — a free text messaging app listed as having between 10-50 million users. We tested 10 other

App	Installs	Location Ads
The Chive	5-10M	✓
Grindr	10-50M	✓
iFunny	10-50M	
Imgur	5-10M	
MeetMe	1-5M	✓
My Mixtapez Music	10-50M	✓
Talkatone	10-50M	✓
TextFree	10-50M	✓
TextMe	10-50M	✓
TextPlus	10-50M	✓
Words with Friends	50-100M	

Table 1: Apps we actively tested. These are the most popular apps among those our DSP could serve ads to.

popular apps to ensure we could also serve targeted ads to them (although not all allowed location targeting), see Table 1.

Experimental Actions. Our devices can be in either an *active* or *inactive* state: in the *active* state the app is open and the device is awake; in the *inactive* state the app is in the background and no ads were being loaded.

Ad Creation and DSP Use. Since we were advertising on behalf of our organization, we obtained approved static image ad content. Thus, when we served these ads to the general populace we were simply conducting real advertising for our organization. When we needed to create numerous ads (as in the location attacks described later) we used the Sikuli automation tool [27] to automate the creation of ads. While we performed very odd targeting compared to a normal advertiser, we never received any negative communication from our DSP over the three-month period we used them.

4.3 Benchmark Evaluations

We begin assessing ADINT capabilities using a series of isolated benchmarks. The goal of these benchmarks is to understand the exact characteristics of our DSP for different operational aspects in a controlled setting. E.g., how long must an app be open to receive our ad, how much will our ad cost to serve, or how precisely can we target ads geographically. We will then use this information to construct our real attacks in the following section.

Benchmark 1: Delay to Service. We first performed two benchmarks: (1) how long the delay is between activating an advertising campaign and the first ads actually being served. (2) how long it takes from an ad being served until our DSP reporting interface shows it was served.

We first activate the advertisement and then enter the active state for the user, timing how long it takes to receive our ad. We then time how long it takes for the ad to be reported as served by our DSP's reporting interface. We perform this benchmark for each user 10 times and show the distribution of times in Table 2: on average a campaign served its first ad within 2m46s, and never took longer than 3m20s. Our DSP reported ads in 6m38s on average, although some took up to 10s longer to be reported.

These benchmarks show that ADINT attacks can be dynamic on a timescale of minutes: new ads, for a new intelligence-gathering campaign, can be active within minutes and the information gained by an ADINT attack can similarly be known within minutes.

	Serve Delay	Report Delay
Mean	2m46s	6m38s
Max	3m20s	6m48s
St. Dev.	0m24s	0m11s

Table 2: Observed delay from campaign activation to first ad serve and from serving an ad to the DSP reporting it.

Benchmark 2: Overall Ad Win Rate and Affordability. How frequently targeted ads will be served — and whether they are affordable to individual actors — is critical to how ADINT can be used by attackers with modest resources. This benchmark examines how often our ad wins its ad auction when the financial investment of the attacker is only moderate.

Our case study DSP, like most DSPs, allows the advertiser to specify the per-impression bid and the ad auction is then run as a second-price auction, so we only pay the second-highest bid [25]. We conducted win rate and cost tests with bids of $0.05, $0.005, and $0.0005 per-impression and then creating ads targeted at our facsimile users, our real user devices, and a set of untargeted ads that could be served to anyone. Testing against real user devices is important to ensure the cost of each ad served is not prohibitively high for ADINT attackers with small budgets. We found our win-rate diminished significantly with bid: $0.05 won 96% of auctions, while $0.005 only won 52%, and $0.0005 only won 15%. However, we found even bidding $0.05 per-impression resulted in paying only $0.005 per-impression on average because of the second-price auction.

When targeted at both real and facsimile user devices, our ads with a bid of $0.05/impression won 90% of auctions and cost no more than $0.02/impression. This means ADINT ads are reliable because they will be consistently served and they are readily affordable to even low-budget attackers. We use the highest bid ($0.05/impression) for all subsequent experiments.

Benchmark 3: First-Ad Dominance. This benchmark measures how often our ad is the first ad served after an app is opened. This is important for tracking when a target visits a particular location because the app may only be open for a short period of time.

To measure this benchmark we enter the active state by opening the app and then wait for the first ad to appear. We record whether this ad was ours or not, and then return to the inactive state for 1 minute. We repeat this cycle 10 times for each user. We also conducted this benchmark test with our real user devices to validate that a potentially richer advertising profile did not cause our ads to be shown less reliably.

We find for real and facsimile user devices that our ad is the first ad 79% of the time. This means we can reasonably rely on our ad being served even when a user only uses an app briefly.

Benchmark 4: Repeat-Ad Dominance. Complementary to the above benchmark, what percentage of ads shown over time in an app are ours is also useful to know for certain attacks. In particular, we could compute how long a user used an app if we know how often ads are fetched and how often our ad is the ad shown, as we demonstrate later in our attacks.

To measure this benchmark we enter the active state for a single user and app for 3 minutes. During the 3-minute period each user sees approximately 16 ads. We record how many ads are served during this time and whether they are our ad or others.

We find for real and facsimile user devices that our ads account for 81% of the ads shown while an app is kept open. This means we can rely on our ad continuing to be served and thus potentially track how long a target has an app open.

Benchmark 5: Location Precision. Our DSP allows "hyperlocal" targeting by inputting GPS coordinates and a radius around them to target ads. Our DSP only allows 4-decimal places of accuracy on these GPS coordinates (approximately 4-11 meter resolution, depending on latitude, which we simplify to 8m) and a minimum radius of 1-meter, so the most precise we can expect this targeting to be is 8m. However, we did not trust that the DSP was necessarily as accurate as its interface claimed. Additionally, smartphone localization can be inaccurate. Therefore, we conducted a series of tests to measure the real world precision of location targeting.

We first recorded network traffic of the app and ad-library and compared the GPS coordinates sent to a ground-truth sample of a GPS app displaying the current location. We found that the ad-library sends the exact same geolocation API coordinates to the advertising ecosystem as the GPS app displays.

To test the precision of actual ad-serving, we created ads targeted at the GPS location of the phone, truncated to 4-decimal places. This ad was always successfully served to all phones. This benchmark does not address the possibility of location ads being inaccurately served to users outside the targeted area: we evaluate this in the next benchmark.

We find that the device's most precise location is transmitted to the ad exchange, and that our DSP does in-practice offer 8m precision, depending on latitude.

Benchmark 6: Location Accuracy. Importantly, the last benchmark does not assess the *accuracy* of the GPS targeting in terms of serving the ad targeted closest to the user and not some other nearby ad. Our sixth benchmark was to evaluate the accuracy of these hyperlocal ads and, in particular, whether ads might also be served to other nearby locations.

We created a grid of hyperlocal ads spaced the minimum distance apart (8m), see Figure 2. We then placed the phones at the same position and waited for a stable GPS location, then entered the active state for three minutes. We observed that 83% of our ads served[1] were to the current phone GPS rounded to the nearest 4-decimal GPS coordinate. The other 17% of cases were an ad targeted at the 4-decimal truncation of a neighboring GPS coordinate. It is unclear why this nondeterminism existed: this occasional error was observed across multiple phones and the GPS coordinates did not appear to change during the experiment.

We find that every hyperlocal ad served was within 8m of the true device location[2], despite also being close to other targeted locations. Thus we can surveil locations at 8m resolution across large spaces by creating these grids of ads.

Benchmark 7: Location Delay. The temporal dimension of location-targeted ads is also important for cases where the user may be in a location for a short amount of time, or for attempting to track a user as they travel from place to place. This benchmark measures two metrics:

[1]Our ads were served 146 times to the 10 phones over three minutes.
[2]While always a location within 8m, a device did not always trigger the *same* hyperlocal ad

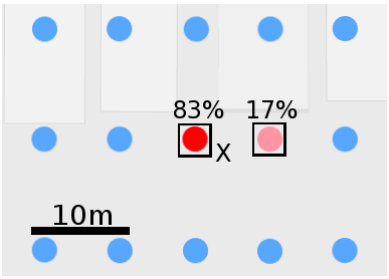


Figure 2: Grid of ads used for testing accuracy. Each dot is an ad, the boxed dots are the ads that were served with the percentage of the trials they were served on, the X marks where the devices were actually located.

(1) How long the user continues receiving location-based ads for a location A after leaving that location;
(2) How long a user must be in a new location B before receiving a location-based ad for that location.

In conducting this measurement we found the two actually had significant impact on each other: if a user moved from some location A to another B and both A and B had ads targeted at them, the user would continue to receive ads for A for between 3-5 minutes after arriving at B. Subsequently they would receive ads for B. If instead the user had not recently received location-targeted ads, then ads for B were shown almost immediately.

We find that tracking a target from one location to another requires them to have been in the new location for 4 minutes. However, serving a location ad to a target not recently location-targeted requires less time, sometimes <1 minute.

4.4 End-to-End Attack Evaluations

Conducting the above benchmarks is important for two reasons. (1) it develops a foundational understanding of the capabilities and limits of using our DSP for ADINT. (2) it allows us to intelligently design end-to-end attacks with confidence, rather than find which attacks are feasible through trial-and-error. We provide several realistically motivated attacks using ADINT below, but also believe our benchmark analysis allows others to more easily assess if other attack ideas are feasible. In each of these attacks, unless noted otherwise, we tested the attack on all 10 facsimile user devices and used Talkatone as the generic app the target opened.

Determine Daily Routine. The objective of this attack is to learn the target's home, office, and frequent hangout locations. We first gather the MAID from the target via sniffing WiFi traffic; we then create a grid of targeted ads around the city the target lives in.

The target then followed a normal daily routine of commuting 2.5 miles to work — 2 miles by bus and 0.5 miles by walking (see Fig. 3 for a map). The target activated their app at least once during their commute while: at home in the morning; walking to the bus stop; at a coffee shop near the bus stop; waiting for the bus; on the bus; walking to the office; and in the office. We conducted this experiment for 7 days with all 10 of the facsimile user devices.

The ads targeted closest to their locations in the home and office were served within the first day in each trial. The ads for the coffee shop and bus stop were only intermittently served — although all were served at least once to each phone within the trial period.

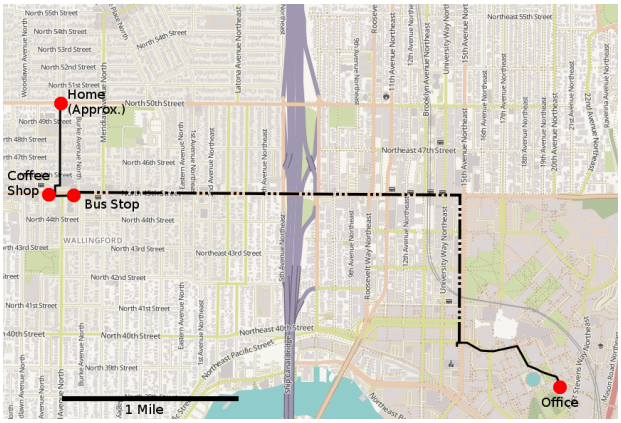


Figure 3: The following map depicts the commute route, proceeding from left to right: red dots are the locations we observe ads targeting (home, coffee shop, bus stop, office); the solid line is walking; the dashed line is busing. To preserve privacy, the location marked as the home is actually an intersection within 0.5 miles of the real location.

This difference can be attributed to the requirement of benchmark 7, location delay: the ad for the coffee shop or bus stop would only have been served if the target stayed there for at least 4 minutes. This also demonstrates why no other intermediate locations during the commute are targeted: the target was constantly moving.

We find that the attacker using the DSP can purchase passive ads and use those ads to determine the home and office locations, as well as any stops longer than 4 minutes, in the movement of a target.

Sensitive Visits. The second location attack we consider is whether we can detect when a target visits a location just once or very infrequently. Example locations of interest might include specialized medical centers, religious centers, known activist meeting points, weapons stores and weapons ranges, corporate offices of a business competitor, and so on. As with the daily routine attack, we begin by sniffing the target's MAID from WiFi traffic. Rather than creating a grid of ads throughout the city, we create a set of five ads, each targeting a 5m-radius point just inside five different buildings on our university campus: these single-point ads are designed to mimic having a set of sensitive locations the attacker hypothesizes the target might go and wait to be served, like those mentioned above.

The target then visited one of the five locations and activated their app in 2 minute intervals, representing idly sitting in a waiting room or lobby. In all cases the ad targeted to that location was served within 10 minutes.

We find that an attacker using our DSP can purchase passive ads and use those ads to detect a target visiting a specific location within 10 minutes of their arrival there.

Crowd Enumeration. A corollary to tracking an individual across locations is tracking unknown individuals at a specific location: e.g., enumerating individuals in a protest at a specific time and location. For this attack, we assume an existing crowd at a certain location. We then create a targeted ad for that location and set the ad to only be served to each device once-per-day (a standard option across DSPs). This way, we can count the number of unique devices being used at that location by simply counting the number of ads served. The precision of our crowd targeting is subject to benchmark 5: location precision, and should be 8m at its maximum precision.

We tested this attack with a simulated crowd of our 10 facsimile users. We found our ad was successfully shown to all of them within 8 minutes (including the 3.3 minutes of ad activation time determined by benchmark 1). Each device only received the ad once, preventing us from over-counting. In practice this attack would likely under-count, since it will only count devices that are in-use and using apps with ads in them.

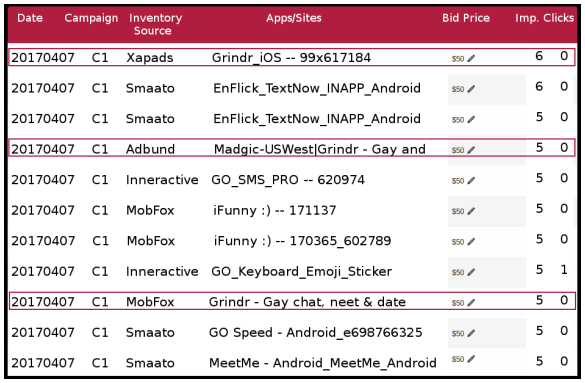
Although using active ad content for ADINT is outside the scope of our case study threat model, this attack could naturally be enhanced by using JavaScript in an ad to set cookies or fingerprint the device for later retargeting. Further, any member of the crowd that clicked the ad would provide their MAID and other fingerprinting features to us automatically. Some users might accidentally click an arbitrary ad, but an ad whose content is pertinent to the members of the crowd (e.g., advertising to donate to a cause associated with the event) might cause a large portion of the participants to click on the ad.

Using the DSP's reporting, an attacker using the DSP can purchase passive ads and use those ads can get a count of devices at the location, when they were there, and some device characteristics such as: gender, device maker, OS, network, and which app the ad appeared in. However, this count is contingent on those devices using apps with ads in them while at the location.

Sensitive App Enumeration. We focused most of our case study on location-tracking as the attacker's intelligence goal, and we did so because location — unlike many other properties — has the potential to be highly dynamic. Having assessed the ability to physically track a target, the ability to learn other sensitive information might be unsurprising. However, we sought to validate that assumption. Here we focus on one such privacy attack: the determination of what apps a target uses. Merely the possession of some apps can be considered sensitive information for a variety of reasons: pregnancy trackers, depression journals, psychiatric drug conditions, and diabetes trackers can all indicate health conditions; dating apps can indicate relationship or sexual preferences; religious text and prayer apps can indicate religion and devoutness.

We begin by sniffing the target's MAID from WiFi traffic, and then create ads targeted at that MAID. Because of the reporting features of our DSP, we do not even need to guess at specific sensitive apps: we simply serve a variety of ad content (to ensure our ad could be shown in any app) and observe what our DSP reports to us about which app our ads were served in (see Fig. 4). We use the Grindr app, which is clearly sensitive to possess in some circumstances [20, 23], on each of the facsimile users to verify this attack works.

While the only sensitive apps we tested was Grindr, we provide a list of other potentially sensitive apps our DSP can serve ads to in Table 3. Similarly, web ads also report what website they were shown on. Our case study focuses on in-app advertising, but website visits could certainly be just as sensitive to users as what apps they have installed.



Date	Campaign	Inventory Source	Apps/Sites	Bid Price	Imp.	Clicks
20170407	C1	Xapads	Grindr_iOS -- 99x617184	$50	6	0
20170407	C1	Smaato	EnFlick_TextNow_INAPP_Android	$50	6	0
20170407	C1	Smaato	EnFlick_TextNow_INAPP_Android	$50	5	0
20170407	C1	Adbund	Madgic-USWest|Grindr - Gay and	$50	5	0
20170407	C1	Inneractive	GO_SMS_PRO -- 620974	$50	5	0
20170407	C1	MobFox	iFunny :) -- 171137	$50	5	0
20170407	C1	MobFox	iFunny :) -- 170365_602789	$50	5	0
20170407	C1	Inneractive	GO_Keyboard_Emoji_Sticker	$50	5	1
20170407	C1	MobFox	Grindr - Gay chat, neet & date	$50	5	0
20170407	C1	Smaato	GO Speed - Android_e698766325	$50	5	0
20170407	C1	Smaato	MeetMe - Android_MeetMe_Android	$50	5	0

Figure 4: Cropped screenshot of the report page; each impression lists what inventory it came from: e.g., "Grindr_iOS", "Grindr - Gay chat, meet & date" and "Madgic-USWest|Grindr" all correspond to an ad being served in the Grindr app.

Gay Dating Apps	
Grindr	Hornet
Jack'D	Romeo
Wapa	Wapo
Dating Apps	
Meet24	MeetMe
Moco	Tagged
Torrenting	
BitTorrent	FrostWire
uTorrent	
Other	
Adult Diapering Diary	Hide My Texts
Hide Pictures Vault	Pregnant Mommy's Maternity
Psiphon	Quran Reciters

Table 3: Example Potentially Sensitive Apps

Additionally, variants of this type of attack that do not require a specific user's MAID are also feasible. E.g., discovering if anyone (or how many people) in a certain area are using certain apps. This is similar to the crowd enumeration attack (above), but extends it to examining what apps individuals in an area are using. The inverse objective — finding where the users of certain apps are — can also be performed by simply creating a grid of location ads targeted at those apps and observing which ones are served. To avoid violating real users' privacy we did not evaluate these attacks. For example, even testing with our facsimile user phones in our building could expose to us the number of real Grindr users in our building.

We find our targeted ads are shown in a sensitive app (Grindr) immediately after the activation time. These are then reported back to the DSP as being served in that context, informing us that the target has the sensitive app (Grindr) installed.

App Usage. To complete our study of potentially non-obvious information that an attacker can extract about a target, we now turn to app usage behaviors. In addition to enumerating apps installed, ADINT can be used to discover when and for how long a target uses an app. We begin the attack by sniffing the target's MAID from WiFi traffic and then creating ads targeted at that MAID.

To test this attack on our facsimile users we activated the app for one of three different durations: 30 seconds, 3 minutes, and 5 minutes. We then attempted to use the resulting report of ads served to calculate when the user began using the app and for how long. We estimated how long the app was used by multiplying the number of times our ad was shown by the refresh-rate (11 seconds for Talkatone) and the reciprocal of how often the ad served is expected to be ours (81% of the time, based on benchmark 4):

$$Time = AdImpressions AdDominance * RefreshRate$$

Real Usage	30s	180s	300s
Avg. Estimated Usage	36s	172s	294s
Std. Deviation	7.75s (25.8%)	23.7s (13.2%)	29.8s (9.9%)
Max. Abs. Error	15s (50.0%)	60s (33.3%)	90s (30.0%)

Table 4: Actual time the app was open and our estimation of the time based on the number of our ads we served and an average of serving our ad 80% of the time any ad is served.

The difference in actual in-app time vs. estimated time is shown in Table 4; we generally see that accuracy is poor for predicting short usage sessions (30 seconds), but grows increasingly accurate for longer sessions (3 and 5 minutes).

We find that we can know how long an app was open to within 20% of actual usage time with 95% confidence when apps are open for 5 minutes.

By targeting multiple users, for example, it might also be possible to use this method to infer information about who is talking with whom (based on whether two users are using the app at similar times, over a sufficiently long duration). An attacker's ability to infer this information is directly related to the public debate about whether communications metadata (information about who is talking with whom, and when) should be private or not [8].

Other Target Information. Our DSP also offers advertisers the option to target ads at specific demographics or user interests, which could be valuable to certain attackers. We chose not to purchase and experiment with demographic- and interest-based targeted ads for several reasons. First, demographics and interests, unlike location, are often less dynamic, and hence many of our benchmarking questions do not apply. Second, and most importantly, is that any evaluation would have required sending demographic- or interest-based targeted ads to either real or facsimile users. Sending such ads to our real users could have compromised their privacy, which we did not want to do. To test with our facsimile users, we would have first had to populate our facsimile user profiles with faux demographic and interest information. Given the financial incentive of DSPs to have high confidence in their ability to deliver targeted ads to users, we posit that our evaluation of targeted ads sent to these faux profiles would largely be an evaluation of our ability to create faux profiles, and not an evaluation of ADINT as a vehicle to learn demographic and interest information about targets.

4.5 Case Study Summary

Our case study provides a systematic evaluation of a canonical DSP for use in ADINT attacks. Our benchmarks demonstrate the baseline capabilities and the attacks we build upon these benchmarks demonstrate that an ADINT attacker can use this DSP to perform a variety of attacks. These attacks include determining:

- How many users are in a location
- Locations a target visits (even only once)
- When a target is in a location

- What apps a target has installed
- When and how long apps are used by a target

These results answer our three driving research questions: (1) *possibility* — privacy attacks using ADINT are possible; (2) *capabilities* — attackers can learn fine-grained location data and sensitive personal data like installed apps and when they are used; (3) *operational aspects* — attackers require only $1000 and a website. We now turn to expanding on these answers by stepping back and evaluating a wider swathe of the advertising ecosystem.

5 SURVEY OF DSPS

The preceding section provided an in-depth case study evaluating several types of privacy attacks using ADINT. We now step back and evaluate the DSP landscape by conducting a survey of capabilities and limitations of 21 DSPs. This survey of DSPs demonstrates that the features of our case study DSP are prevalent in the ecosystem, i.e., that other DSPs could have been used for the same attacks explored in our case study. Our survey also shows new and different ADINT capabilities not explored in our case study, such as targeting individuals by PII (name, email, and physical address).

We selected the 20 additional DSPs from several sources: a previous examination of four DSPs for use in unconventional display advertising by Zimmerman [30]; three general Internet companies that offer advertising (Bing, Facebook, and Google); and other DSPs representing a variety of costs and specializations. Table 5 shows the features of these DSPs.

Methodology. We contacted each DSP with a series of questions about its capabilities between November 2016 and May 2017. If the DSP was free, we explored its capabilities by creating an account and using the service. Otherwise, we participated in a guided demo by a sales representative. We were careful not to mislead the DSPs we contacted, and our organization was interested in using our results to inform future advertising decisions for itself.

5.1 Targeting Criteria

As shown by our case study, the targeting criteria a DSP allows are critical to its use in ADINT: these criteria provide the fundamental limits of what kind of information an attacker can obtain. Table 5 shows that targeting criteria can vary greatly between DSPs.

DSPs have developed new features over time to create more value for their advertiser customers. One example is Facebook: in 2010 Korolova conducted early work on targeted advertising privacy threats, but could only target individuals by finding unique combinations of basic demographic characteristics and distinctive "Likes" [15]. Now, however, Facebook has added the ability to upload lists of email addresses or phone numbers to target matching individuals. DSPs continue to push the envelope in a number of areas, such as linking users across devices and linking offline actions to online users.

In order to create a useful resource for future ADINT research, we aggregated the raw capabilities into broader categories of targeting. A "✓" represents a common baseline capability in an area (e.g., targeting age within demographics is common to every DSP offering demographic targeting). If a DSP has more extensive features it is denoted with a "+".

Demographics. 75% of our DSPs provide the baseline demographics options: targeting based on age, gender, and language

35% of our DSPs provided more extensive demographic targeting. This included features such as race (AdWords, BluAgile, Centro, Facebook, MediaMath), sexual preference (Adwords, Facebook), finances (BluAgile, Centro, Choozle, ExactDrive, Facebook), homeownership (Facebook), political affiliation (AdWords, Choozle, Facebook), and employer (Centro, Choozle, Facebook).

Interests. 80% of our DSPs allow interest-based targeting. The baseline case is using the 392 different interests from the Interactive Advertising Bureau (IAB) [4]. Most of these are innocuous, but some — such as A.D.D., AIDS/HIV, heart disease, incest support, incontinence, immigration, legal issues, various religious categories, or U.S. military careers — could certainly be sensitive.

40% of our DSPs allowed targeting more advanced interests. These included pages, articles, or groups liked on Facebook or on- and offline purchases (Choozle, Facebook, GetIntent, MediaMath).

Personally Identifiable Information. 40% of our DSPs allowed targeting ads based on information that is generally unique to individuals. All of these cases supported targeting users based on email addresses. Some services required minimum numbers of emails to use: Facebook required 20+ emails, AdWords required 1000+ emails, and Centro requires 100,000+ emails. The other DSPs (Admedo, AdRoll, Choozle, MediaMath, MightyHive, Tapad), however, did not list minimums. Of course, as the work by Korolova that prompted Facebook to institute a minimum noted [15], these minimums can be circumvented; we conducted a preliminary experiment and found uploading 19 entirely spurious email addresses (not even connected to fake Facebook accounts) allowed us to target ads at a test user.

Some DSPs also allow targeting based on other PII: Facebook allows targeting based on phone numbers; MightyHive supports targeting based on names and physical mailing addresses.

Cookies/MAID. Every DSP allows targeting users based on cookies or mobile advertising ID (MAID). Either of these could be obtained by an ADINT attacker if the user ever clicks on their ad. They can also be obtained from sniffing network traffic. Finally, active ad content (see below) can be used to potentially acquire either identifier.

Device. 85% of our DSPs support targeting based on device properties. These properties include the browser brand (in the case of web ads), the operating system type, and the device type (i.e., phone, tablet, etc.). In some cases the exact operating system version (Centro, Facebook, LiquidM, MightyHive, Splicky) and specific device make and model (Centro, Facebook, MightyHive, SiteScout, Tapad) can be specified.

Network. 85% of our DSPs allow targeting based on network characteristics. The basic version of this is targeting devices on cellular versus WiFi networks. 35% of the DSPs also allow arbitrary IP white- and blacklisting (Admedo, AdWords, Bing, BluAgile, Criteo, Centro, Choozle, Go2Mobi, Simpli.fi).

IP targeting could be used for ADINT in several ways: it can act as a proxy for location if the IP address for a particular company, home, or open WiFi network is previously known. IP can also stand in as a semi-unique identifier if something more specific like a cookie

DSP	Min. Cost	Targeting									Content			
		Demographics	Interests	PII	Cookie/MAID	Device	Network	Location	Domain/App	Search	HTML	Flash	3rd-Party	Web Beacon
Case Study	$1,000	✓	✓	-	✓	✓	✓+	✓+	✓	-	✓	-	✓	✓
Admedo	$5,000	✓	✓+	✓	✓	✓	✓+	✓+	✓	-	✓	✓	✓	✓
AdRoll	$0	-	-	✓	✓	-	-	-	-	-	-	-	-	-
AdWords	$0	✓+	✓+	-	✓	✓	✓+	✓+	✓+	✓	✓	✓	-	✓
Bing	$0	✓	✓	-	✓	✓	✓	✓	✓	✓	-	-	-	-
Bonadza	$300	✓+	-	-	✓	✓	-	✓+	✓	-	✓	-	✓	✓
BluAgile	$1,000	✓+	✓	-	✓	✓	✓+	✓+	✓	-	✓	✓	✓	✓
Centro	$5000 / month	✓+	✓+	✓	✓	✓	✓+	✓+	✓	✓	✓	✓	✓	✓
Choozle	$99 / month	✓+	✓+	✓	✓	✓	✓+	✓	✓	-	✓	✓	✓	✓
Criteo	$0	-	-	-	✓	-	✓	✓	-	-	✓	✓	✓	✓
ExactDrive	$50	✓	✓	-	✓	-	✓	✓+	✓	✓	✓	✓	✓	✓
Facebook	$0	✓+	✓+	✓	✓	✓	✓	✓+	-	-	-	-	-	-
GetIntent	$0	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Go2mobi	$0	-	-	-	✓	✓+	✓+	✓	✓+	-	✓	✓	✓	✓
LiquidM	$1,000	✓	✓	-	✓	✓	✓	✓+	✓	-	✓	-	✓	✓
MediaMath	$50,000 / month	✓+	✓	✓	✓	✓	✓	✓+	✓	-	✓	✓	✓	✓
MightyHive	$2,000	✓	✓	✓	✓	✓	✓	✓+	✓	✓	✓	✓	✓	✓
Simpli.fi	$10,000	✓	✓+	-	✓	✓	✓+	✓+	✓	✓	✓	✓	✓	✓
SiteScout	$500	-	✓+	-	✓	✓	-	✓	-	-	-	✓	-	-
Splicky	$0	✓	✓	-	✓	✓	✓	✓+	-	-	-	-	-	-
Tapad	$2,000	✓	✓+	✓	✓	✓	✓	✓+	✓	✓	✓	✓	✓	✓

Table 5: Summary of DSP features: targeting features can be used by ADINT operators for gathering information; different content types can enable the actual ads content to gather even more information on behalf of an ADINT operator.

or MAID cannot be used. Finally, IP can be used to link devices or individuals that frequently share the same NAT'd network.

Location. All but one of our DSPs allows some form of location-based targeting. The basic version of this (just a "✓") is restricted to city or ZIP code level of granularity.

60% of our DSPs provide more advanced location targeting — typically termed "hyperlocal" — which varies significantly in granularity. AdWords and Facebook only allow targeting radii of 1-mile. Both also allow exclusionary radii, but it is unclear how granular of targeting this actually allows.

Other DSPs claim 1-meter radius granularity (Bonadza, Centro, LiquidM, MediaMath, Simpli.fi, Splicky, Tapad), although as our case study shows this does not necessarily mean 1-meter accuracy is possible in practice.

Domain/App. 75% of our DSPs allow the advertiser to specify which context — domains (for web ads) or apps (for mobile ads) — their ads will appear in. In Adwords and Centro this specification can include particular pages within a domain. Restricting ads by domain or app could be used to find people that read certain websites or use certain apps. Furthermore, the reporting of websites and apps could allow an ADINT attacker targeting a particular individual to see what websites they visit or apps they use — as shown with sensitive apps in our case study.

Search. 40% of our DSPs supported using search keywords for targeting. Search terms can also be indirectly targeted via interests because the interest profile of a user is partially populated based on what they search for. However, these DSPs are those that specifically allow search keywords to be targeted with ads. Search targeting could be used in a manner similar to targeting domains and apps.

5.2 Ad Content

Our case study evaluation focused on ad targeting, but the content of ads can be combined with targeting to enhance ADINT capabilities. This makes the variance in ad content among DSPs another feature of interest to compare. 80% of DSPs will serve either Flash or HTML5 ad content. These types of active content allow the ADINT attacker to execute potentially arbitrary JavaScript as part of their ad. This capability could allow an attacker to fingerprint a target device, set cookies, or extract the MAID [22]. Most DSPs specify these types of ads are audited, but this process varies: AdWords has a fully-automated framework which could be tested against by an attacker; Choozle stated there was no auditing procedure at all. In addition, 70% of DSPs allow the use of third-party hosting for ad content, where the advertiser hosts the actual ad content themselves or with a dedicated hosting provider. This arrangement means the DSP no longer has control over the ad content, so it becomes even easier to use the ad content for ADINT purposes.

Web beacons are another important feature of ad content. Web beacons send a request when an ad is loaded, independent of any user action. The request allows the advertiser to learn the IP address, User Agent String, and Cookie, of the device seeing the ad. This can allow easy reidentification of the device from previous ADINT actions, as well as potentially provide new information, such as if their IP address changed.

5.3 DSP Survey Summary

Our survey demonstrates that our case study DSP is hardly a unique case. If our case study has shown ADINT is possible when using that DSP, then our survey shows these operations and even more are possible using these other DSPs.

As can be seen in Table 5, DSPs can vary greatly from one another across targeting criteria, content options, and costs. One of the rarest capabilities is targeting based on PII, such as uploading a list of emails or real names to a DSP for targeting. PII-targeting would be the easiest way to start targeting a specific individual (more about these approaches in Section 6). However, the other targeting capabilities of these DSPs are often lesser — only Centro and MediaMath offer both PII and hyperlocal targeting, and at $50,000/month MediaMath may fall outside the resources of some ADINT attackers. Search-targeting capabilities are similarly rare.

Across ad content there are few DSPs with restrictions: only Facebook, AdRoll, and Splicky restrict ad content to static images and text. Thus there is considerable opportunity to couple active ad content with advanced targeting options for increased ADINT capabilities, a potentially rich area of future work.

6 DISCUSSION

As the preceding two sections demonstrated, the concept of ADINT enables a wide range of privacy attacks that are practical for even modestly-resourced individuals to conduct. Here we explore some of the potential attackers and discuss potential defenses.

6.1 Potential ADINT Attackers

The reason ADINT is important to research is that it takes data that many people know impersonal corporations already have, and shows that arbitrary individuals with mixed motives can access that data as well. The attacks conducted in our case study most closely resemble the goals of a traditional stalker. However, we postulate many other types of attackers would find value in ADINT attacks. Below we describe a few such attackers and how they could use ADINT both to find targets from the general populace and then extract sensitive data about them.

Ideological Vigilantes. Across the globe there are many instances of individuals and groups organized with the objective of enforcing cultural norms or ideological positions on members of their community. These types of groups can range greatly in objectives, violence of actions, size, and resources. However, all of them typically act by finding individuals that violate the group's ideology and concentrating their force on that individual.

Depending on the types of targets they are seeking, the targeting criteria provided by ad-networks could directly facilitate finding targets. E.g., an anti-gay group could conduct target acquisition by serving ads in gay apps or location-targeting gay bars and extracting identifiers. That information alone could be sufficient for the group's purposes, e.g., if that information exposed the number of gay people at a specific location. The group could also use ADINT to gather more information about the targets prior to carrying out some other nefarious objectives.

Criminals. Stalkers, burglars, and blackmailers, can all make use of ADINT's democratization of intelligence. Depending on how a stalker chooses their victims, a variety of target acquisition criteria could be used, including frequenting the same coffee shop or having a particular ethnic background. Once a target is acquired, a stalker might closely parallel the very attacks we performed in our case study, such as finding a victim's home, office, and hangouts.

A burglar might select targets by financial categories or past purchases of luxury goods, and then use location attacks to find where the target lives and ensure they are away at the time of the crime. Blackmailers could similarly use financial targeting to find worthwhile victims and then further targeting to gather exploitable intelligence: e.g., knowing when the victim visited a brothel.

Business. We also envision numerous business-related use cases for ADINT, ranging from media-related uses to financial investing. A paparazzi might send targeted ads to the home locations of celebrities using a DSP like Choozle that supports PII-based targeting. The delivery of those ads could leak sensitive information about those celebrities to the paparazzi, such as what apps they use or what interests they have. The paparazzi could also use those initial ads, with active content, to learn the celebrities' MAIDs, after which they could track those celebrities over time. Other journalists might use similar techniques to glean information about politicians and other high-profile individuals.

Financial investors might, instead, acquire identifiers for venture capitalists or executives of companies using PII targeting. Subsequently using the location targeting capabilities from our case study, it could be possible to determine when the VCs or executives visit other companies, possibly indicating when there might be a large round of funding, acquisition, or big announcement, thereby providing valuable investment-related information.

6.2 Anatomy of an ADINT Operation

Adapting tools designed for targeted advertising to intelligence gathering requires some additional indirection that is not necessary when using purpose-built intelligence collection tools. Below we provide an abstracted pipeline of how we envision that a complex ADINT operation might generally proceed. This pipeline applies to ADINT operations trying to deeply surveil a target, like most of our case study attacks; it is not as applicable when the attacker is conducting simpler one-time attacks, such as crowd enumeration (see Section 4.4). The pipeline consists of three stages: (1) target acquisition, (2) strengthening identifiers, and (3) targeted surveillance, which we describe below.

Target Acquisition. The target acquisition stage of an ADINT operation is for attackers that know *what kind* of targets they want to surveil, but do not have identifiers for those targets. The more in-depth attacks demonstrated in our case study — like tracking the user in the physical world or finding sensitive apps they have installed — require specific identifiers to extract information, so an attacker needs to obtain identifiers for their targets to conduct these more powerful attacks.

In our case study, this stage involved sniffing the MAID of our targets from network traffic. However, in other ADINT operations a number of different techniques could be used: e.g., obtaining an IP address of a target can be done with a simple Web Beacon and no user interaction; more sophisticated ad content or convincing the user to click on an ad can convey even stronger identifiers (like setting cookies or extracting the MAID).

Using ads for target acquisition allows the targeting features discussed in Section 5 to be used. This gives ADINT attackers the ability to turn abstract target criteria — such as being a member of

a specific religion, using a specific app, or being at a specific location — into a list of identifiers for targets matching that criteria. For example, the attacker could use location-targeted ads to a specific government building to learn the IP addresses (with Web Beacons) or the MAIDs (with more sophisticated ad content) of those inside.

Strengthening Identifiers. The second stage is strengthening the identifiers generated by target acquisition. This may be necessary because of the variance in targeting and content capabilities of different DSPs. For example, an attacker might want to know every user that visits a particular webpage. AdWords allows very fine-grained web-page targeting but also has stricter content auditing that might restrict an ad to only being able to extract the IP address of a user. Thus, in this example the attacker is only able to obtain IP addresses in the target acquisition phase; this does not allow location tracking, since the target's IP address, if connected to WiFi, will change as they move between networks.

To overcome this limitation, the attacker can use a DSP with laxer auditing, like Choozle, to retarget the IP addresses with ads that extract a MAID. Then the attacker can proceed to use the MAID, a stronger identifier, to perform location tracking attacks with a DSP like ou case study DPS.

Targeted Surveillance. The third stage is using targeting identifiers in ADINT operations to learn new facts about the targets. As shown by our case study and survey in Sections 4 and 5, there are many different types of information an attacker can learn using ADINT. Our case study focused on physical location and app usage as information that we could rigorously test on facsimile users. However — as our survey of DSPs shows — the space of information that DSPs can target is much larger and touches on many different aspects of targets. For example, various personal details like sexuality, religion, finances, and search terms can all be targeted.

6.3 Defenses

There are several different ways in which ADINT attacks can be mitigated. Users could avoid many ADINT attacks by never using apps or visiting websites with ads. However, entire segments of apps and websites operate solely by selling advertising, so this is not necessary practical. Some of the most sensitive user data could be kept hidden by a user, but this could prevent use of some services: e.g., many dating apps need location, and may want sensitive data to better match their users. Constantly resetting device identifiers like cookies and the MAIDs also provides some protection. However, this poses a significant burden on users to continually reset identifiers. Furthermore, in our study we found that ads targeted at a particular MAID continued to be served to the device up to 6 hours after resetting that MAID. Depending on how the attacker obtains the MAID, this may allow them to acquire the new MAID.

Preventing ADINT appears more feasible from the ad-network perspective. Some ad-networks have already done this: Facebook and Google both have thresholds on how few users an ad can specify that it targets (20 and 1,000, respectively). However, both of these ad-networks are abnormal in the ecosystem because they are also software platforms with large user-bases. Thus, they have a market incentive to protect user privacy that many other ad-networks do not. Furthermore, these thresholds make ADINT more challenging, but do not prevent it, since spoofed or sybil accounts can be used to circumvent them.

Heuristics and data-driven approaches to catch ADINT attackers based on odd advertising activity is another potential method for stopping ADINT. However, for ad-networks without direct user-bases it is unclear there is any market incentive to deploy these defenses. Given this is a problem of market incentive, regulations requiring these kinds of defenses to conduct advertising business within a country could be effective. These defenses are not perfect solutions but they could greatly increase the difficulty of ADINT attacks.

7 CONCLUSIONS

In this work we explored the concept of ADINT — use of targeted advertising to conduct privacy attacks on users. We evaluated the capabilities of ADINT in a case study of location and app surveillance with one DSP. We additionally surveyed 21 DSPs to evaluate the variety of capabilities and costs they have for use in ADINT and demonstrate the potential for many more types of attacks. Finally, we explored the impact ADINT attacks could have and potential defenses against them. Given the potential privacy implications of ADINT, its capabilities, and its ease of use by low-budget adversaries, we encourage additional research discussions around ADINT, not just within the computer security community, but within the policy and regulatory communities as well.

ACKNOWLEDGMENTS

This work was supported in part by NSF Award CNS-1463968, the University of Washington Tech Policy Lab, and the Short-Dooley Professorship. We kindly thank our anonymous reviewers for their helpful feedback. Finally, thanks to Ryan Calo, Sandy Kaplan, Kiron Lebeck, Peter Ney, Lucy Simko, and Anna Kornfeld Simpson for helpful discussions and reviews of earlier drafts of this paper.

REFERENCES

[1] Gunes Acar, Christian Eubank, Steven Englehardt, Marc Juarez, Arvind Narayanan, and Claudia Diaz. 2014. The Web Never Forgets: Persistent Tracking Mechanisms in the Wild. In *Proceedings of the ACM Conference on Computer and Communications Security*.

[2] Rebecca Balebako, Pedro Leon, Richard Shay, Blase Ur, Yang Wang, and L Cranor. 2012. Measuring the effectiveness of privacy tools for limiting behavioral advertising. In *Web 2.0 Security and Privacy*.

[3] Hal Berghel. 2001. Caustic Cookies. (2001). http://www.berghel.net/col-edit/digital_village/apr-01/dv_4-01.pdf.

[4] Interactive Advertising Bureau. 2015. IAB Tech Lab Content Taxonomy. (2015). https://www.iab.com/guidelines/iab-quality-assurance-guidelines-qag-taxonomy/.

[5] Interactive Advertising Bureau. 2017. IAB Interactive Advertising Wiki. (2017). https://wiki.iab.com/index.php/Category:Glossary.

[6] Giuseppe Cattaneo, Giancarlo De Maio, Pompeo Faruolo, and Umberto Ferraro Petrillo. 2013. A review of security attacks on the GSM standard. In *Information and Communication Technology-EurAsia Conference*. Springer, 507–512.

[7] Robert M Clark. 2013. Perspectives on Intelligence Collection. *The intelligencer. Journal of US Intelligence Studies* 20, 2 (2013), 47–53.

[8] David Cole. 2014. We kill people based on metadata. *The New York Review of Books* 10 (2014), 2014.

[9] Jonathan Crussell, Ryan Stevens, and Hao Chen. 2014. Madfraud: Investigating ad fraud in android applications. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. ACM, 123–134.

[10] Doug DePerry, Tom Ritter, and Andrew Rahimi. 2013. Cloning with a Compromised CDMA Femtocell. (2013). https://www.defcon.org/images/defcon-21/dc-21-presentations/DePerry-Ritter/DEFCON-21-DePerry-Ritter-Femtocell-Updated.pdf.

[11] Google Developers. 2017. Google Ads. (2017). https://developers.google.com/ads/.

[12] Steven Englehardt and Arvind Narayanan. 2016. Online tracking: A 1-million-site measurement and analysis. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 1388–1401.
[13] Steven Englehardt, Dillon Reisman, Christian Eubank, Peter Zimmerman, Jonathan Mayer, Arvind Narayanan, and Edward W Felten. 2015. Cookies that give you away: The surveillance implications of web tracking. In *Proceedings of the 24th International Conference on World Wide Web*. ACM, 289–299.
[14] Go2mobi. 2017. (2017). https://www.go2mobi.com.
[15] Aleksandra Korolova. 2010. Privacy violations using microtargeted ads: A case study. In *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*. IEEE, 474–482.
[16] Zhou Li, Kehuan Zhang, Yinglian Xie, Fang Yu, and XiaoFeng Wang. 2012. Knowing your enemy: understanding and detecting malicious web advertising. In *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM, 674–686.
[17] Nicolas Lidzborski. 2014. Staying at the forefront of email security and reliability: HTTPS-only and 99.978 percent availability. (2014). https://googleblog.blogspot.co.uk/2014/03/staying-at-forefront-of-email-security.html.
[18] Steve Mansfield-Devine. 2015. When advertising turns nasty. *Network Security* 2015, 11 (2015), 5–8.
[19] Jeffrey Meisner. 2014. Advancing our encryption and transparency efforts. (2014). https://blogs.technet.microsoft.com/microsoft_on_the_issues/2014/07/01/advancing-our-encryption-and-transparency-efforts/.
[20] Rick Noack. 2014. Could using gay dating app Grindr get you arrested in Egypt? (2014). https://www.washingtonpost.com/news/worldviews/wp/2014/09/12/could-using-gay-dating-app-grindr-get-you-arrested-in-egypt/.
[21] Franziska Roesner, Tadayoshi Kohno, and David Wetherall. 2012. Detecting and Defending Against Third-Party Tracking on the Web. In *USENIX Symposium on Networked Systems Design and Implementation*.
[22] Sooel Son, Daehyeok Kim, and Vitaly Shmatikov. 2016. What mobile ads know about mobile users. In *Proc. 23rd Annual Network and Distributed System Security Symposium (2016)*.
[23] Mark Joseph Stern. 2016. This Daily Beast Grindr Stunt Is Sleazy, Dangerous, and Wildly Unethical. (2016). http://www.slate.com/blogs/future_tense/2016/08/11/the_daily_beast_s_olympics_grindr_stunt_is_dangerous_and_unethical.html.
[24] Ryan Stevens, Clint Gibler, Jon Crussell, Jeremy Erickson, and Hao Chen. 2012. Investigating user privacy in android ad libraries. In *Workshop on Mobile Security Technologies (MoST)*. 10.
[25] Ratko Vidakovic. 2013. The Mechanics Of Real-Time Bidding. (2013). http://marketingland.com/the-mechanics-of-real-time-bidding-31622.
[26] Craig E. Wills and Can Tatar. 2012. Understanding what they do with what they know. In *ACM Workshop on Privacy in the Electronic Society*.
[27] Tom Yeh, Tsung-Hsiang Chang, and Robert C Miller. 2009. Sikuli: using GUI screenshots for search and automation. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*. ACM, 183–192.
[28] Apostolis Zarras, Alexandros Kapravelos, Gianluca Stringhini, Thorsten Holz, Christopher Kruegel, and Giovanni Vigna. 2014. The dark alleys of madison avenue: Understanding malicious advertisements. In *Proceedings of the 2014 Conference on Internet Measurement Conference*. ACM, 373–380.
[29] Tiliang Zhang, Hua Zhang, and Fei Gao. 2013. A Malicious Advertising Detection Scheme Based on the Depth of URL Strategy. In *Computational Intelligence and Design (ISCID), 2013 Sixth International Symposium on*, Vol. 2. IEEE, 57–60.
[30] Peter Thomas Zimmerman. 2015. *Measuring privacy, security, and censorship through the utilization of online advertising exchanges*. Technical Report. Tech. rep., Princeton University.