

# Summary of “An Investigation into Android In-App Ad Practice: Implications for App Developers”

Stephane Meunier

*Department of Mathematics and Computer Science*

*Clark University*

Worcester, Massachusetts

smeunier@clarku.edu

Catalin Veghes

*Department of Mathematics and Computer Science*

*Clark University*

Worcester, Massachusetts

cveghes@clarku.edu

**Abstract**—The purpose of this paper is to summarize the work [1] conducted by Boyuan He, Haitao Xu, Ling Jin, Guanyu Guo, Yan Chen, and Guangyao Weng during their research which was published in IEEE INFOCOM 2018 Conference.

In-app advertising represents the major source of income for millions of developers and application owners. To monetize their mobile apps, these people commonly utilize various ad networks and ad types to generate revenues in order to sustain their products. Even though it is a widespread habit, especially among free applications which do not charge users for downloading the software, there are neither formal studies nor recommendations about this segment’s best practices. The conference paper under discussion strives to reveal suitable implementations for advertisements and provide stakeholders guidelines on ad network selection and ad placement to maximize income without affecting the overall user experience. While working on this paper, the authors investigated 277,616 Android apps and extracted 697 unique APIs belonging to 164 ad networks. Based on the similarities they observed between different advertisements, the team was able to classify them into five ad types: Embedded, Popup, Notification, Offerwall, and Floating. These are the basic categories which are being used throughout the research in order to concisely articulate their findings among a significant number of alike yet different kinds of advertisements.

**Index Terms**—in-app ads, Android, mobile applications, API, advertisements, ad networks.

## I. INTRODUCTION

### A. Motivation and State of the Art

The focus of this work is the Android platform which has increasingly become the dominant operating system for mobile devices. Its salient importance is supported by numbers and facts rigorously cited and examined by the authors. Android’s worldwide market share has hit 86.1% in the first quarter of 2017 and the number of apps in Google Play has reached 2.9 million in the February 2017, 95% of which are free apps [2]. As most of the free apps leverage third-party in-app advertising for generating money, having a good understanding of ad networks and their associated ad types is of great importance. According to an economic forecast [3], the in-app advertising market is worth around 62 billion US dollars and significantly affects thousands of people at an economic level. Therefore, by providing clear rules and principles, this research is extremely valuable as in-depth analysis and coherent guidelines are essential for adequate development of the in-app advertisement segment. Although the mobile advertising ecosystem has been

the subject of numerous recent researches, most of the efforts focus on security, the privacy of ad networks, mobile ad fraud, and ad targeting. The current state of the art does not include any published materials that address the challenge tackled in this paper.

### B. Contribution

To provide a reliable set of principles to follow when it comes to choosing best ad networks and appropriate ad types, the paper studies Android in-app advertising from a developer’s perspective. The authors developed a static analysis framework for Android apps called MAdLens. MAdLens was used to automate the tedious task of extracting relevant information from the pool of 277,616 free Android applications, randomly selected from Google Play Store. Leveraging on this system, they further performed a large-scale measurement across the Android market and identified trends in the usage of mobile ad networks, aggressiveness difference of ad types and its impact on the usability of the apps. There are three major contributions made by this research:

- It is the first to provide practical guidelines for stakeholders to monetize their apps through third-party in-app advertising.
- It is the first to map APIs of ad networks to specific ad types and measure third-party in-app advertising in Android apps at API granularity.
- It is the first to provide a unified classification system for mobile in-app ads and measure the impact of different ad types on apps.

### C. Paper Structure

The first Section of their paper includes the abstract and the introduction which present the issue, its significance in the contemporary context, the concepts behind MAdLens, and some notable findings of in-app mobile advertising. Section 2 provides relevant background in Android third-party ads, while Section 3 offers a detailed analysis of MAdLens system as well as a comprehensive discussion of the methodology used by this system. Section 4 demonstrates the result of their large-scale measurements and gives explanations as well as implications based on their results. Section 5 discusses the

limitations of their work. Section 6 presents related work and Section 7 concludes the paper.

## II. RELATED WORKS

In recent years, mobile advertising has become an important topic for research especially because of the security concerns that it poses. Some of the preeminent materials mentioned in this paper study topics like the potential risks of embedded or in-app ad libraries [4], the risk associated with user data exposure to advertising libraries [5], a privacy protection framework to “achieve an equilibrium” between the developer’s revenue and the user’s privacy [6], the mobile ad fraud perpetrated by Android apps [7], the user targeting strategies of top ad networks [8], or the use of machine-learning to detect ad libraries and then use code instrumentation to deescalate their unnecessary privileges [9]. All these materials inspect the amount of sensitive information and the ways ad libraries handle it in order to provide relevant ads to the end users while preserving their individual privacy. For example, one of the studies showed that most existing libraries collect private information for legitimate targeting purposes (such as the user’s location) along with additional data whose purpose is hard to justify (such as the user’s call logs, phone number, browser bookmarks, or even the list of apps installed on the phone). Moreover, some libraries make use of an unsafe mechanism to directly fetch and run code from the Internet, which immediately leads to serious security risks [4]. Even though the issue regarding data security and its integrity in mobile applications is often addressed in academic literature, there is no guideline that helps stakeholders to make decisions about the appropriate ads for their software, besides the existing security recommendations. Therefore, the paper under discussion represents the state of the art in its area.

## III. SYSTEM MODEL

### A. Mobile Ad Network and its Integration in Apps

The current ecosystem of mobile advertising involves three major members: the publishers who own a mobile application and who desire to display ads in their software to make revenue, the advertisers who pay to get their content advertised, and the ad networks which serve as intermediate platforms that bridge the connection between the first two members. To integrate the ad network mechanism developers must include one or more pre-compiled ad libraries provided by ad network companies. These libraries and their necessary dependencies are generally implemented in Java and come in the form of compiled jar files. After importing the libraries, programmers can interact with them using specific APIs. Most ad networks provide a rich API surface which allows developers to customize the visual aspects of the ads they are displaying. Additionally, relevant stakeholders need to create an account on the ad network’s website to set up the payment credentials and individual preferences (suitable ad contents, desired ad frequency, layout, format, and others).

### B. Mobile Ad Classification and Aggressiveness

Advertisements within mobile applications can be displayed in many different forms. The Conference Paper labels each ad type based on its Aggressiveness and its Format. The Aggressiveness of a mobile ad is defined by establishing how invasive and unpleasant it is to the user. For the types that can be ignored, skipped or closed, the article labels their Aggressiveness as being “Low”. On the other hand for those that are more invasive, their Aggressiveness falls into the “High” category. According to the piece of writing, Ad Networks support many different ad formats. Deciding which configuration would suit the best is a quality that any mobile developer should possess. After carefully analyzing distinct ad networks and their different available formats, the authors have decided that classifying the ads would be the most convenient option. The research paper categorizes mobile ads in 5 main different types: Offerwall, Notification, Popup, Floating, and Embedded.

Offerwall is a type of ad that is considered to be one of the most invasive. Offerwall uses a reappearing page that offers to end user in-app rewards in exchange for completing certain tasks (i.e. downloading other apps or watching videos). It often interrupts the app’s behavior and cannot be ignored, skipped or closed. That is why its Aggressiveness is considered to be High.

Notification displays its ad content the same way that a regular app would display a notification on the mobile device. The main difference between this ad format and the others is that it will still be running even when the app is closed. The advantage for the ad networks using this type is that the notification will be displayed using the same ringtone as other apps, deceiving the user to look at the mobile device. On the other hand, the disadvantage is that it can distract the user or even interrupt the use of other apps. That is why Notification type is consider to be Highly Aggressive.

Not all ad types are invasive to the user, the research paper categorizes the following 3 ad types as having Low Aggressiveness.

As its name suggests, Popup type displays a pop-up window in front of the app’s GUI. Popups can be different forms or sizes and they can even take up the full screen if necessary. Although this type of ad could be unpleasant to the user, the study describes its Aggressiveness as Low. Even though Popups may be invasive and interrupt the app’s behavior, they can be easily closed by the user.

Although the Floating type of ad is not very common, it is relevant to the study of different ad Formats. This type of ad appears in front of GUI as a floating bubble. Since it does not occupy a large portion of the screen, and also can be skipped or closed, the research categorizes its Aggressiveness as Low.

The Embedded ad is considered to be the least Aggressive Format among the others. This type embeds advertisement content into a small window inside of the app. It can be displayed in many different ways, such as banner, video and etc. The ads displayed this way usually are very small and do

not interrupt the flow of the app. They can also be ignored, skipped or closed.

#### IV. METHODOLOGY

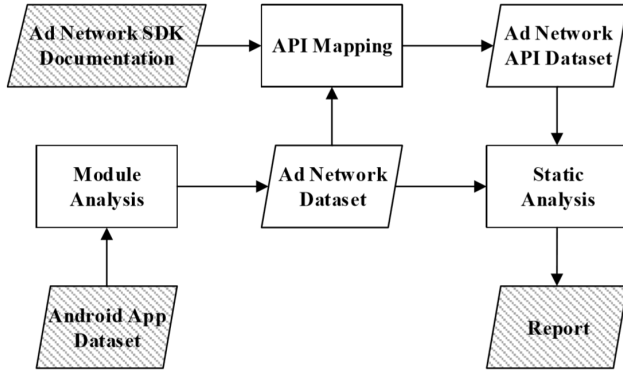


Fig. 1. Overview of MAdLens

The authors created MAdLens system (Fig. 1) which can identify internal third-party ad network modules from an Android app, map APIs of ad networks to specific ad types, and generate a detailed report of all the ad information for each app using static analysis techniques. Part of this system is a crawler that randomly picks applications from Google Play to create the Android App Dataset. For each application in the dataset, MAdLens automatically distinguishes all existing ad networks and then, using the API Mapping created based on Ad network SDK Documentation, classifies the present APIs in one of the five categories defined by the authors and generates an appropriate summary. To extract the relevant information from the applications, the system considers the fact that ads can be implemented either in the resource file (XML) or in Java code. For resources files, they decompile the apk file with Androidguard to get all the Android layout files, then parse all the XML to find the ads. Similarly, for Java code, the smali code is generated by Androguard after decompiling the apk file and ads are recognized in the smali text.

#### V. DATA ANALYSIS AND RESULTS

The research paper revealed multiple useful discoveries regarding mobile ads for Android applications. Developers may want to pay attention to the different implications that the authors have concluded after conducting a large-scale measurement study. There are 8 crucial implications which arose during the analysis:

- Due to the rise of different ad networks, developers have many different choices when it comes to determining ad placement in their apps. However, most developers take a more conservative approach when it comes to ads: 71% of apps contain at most one ad library.
- When it comes to choosing Ad types, programmers tend to choose Low-Aggressive ad types: Embedded and

Popup. They are present in more than half of the analyzed apps. In contrast, the most aggressive types of Ads, Offerwall and Notification, are the least used ad types.

- The number of Ads in an app is directly correlated with the type of category that app belongs to. The research paper affirms that business or utility apps are much less likely to have ads compared to apps that fall into the category of entertainment or casual.
- According to the percentage of apps accommodating at least three types of ads, the research paper concludes that the two slightly aggressive ad types, Embedded and Popup, are the most popular among all the different categories. Although Offerwall is the most aggressive type of ad, it is relatively popular. The paper suggests that developers should consider using these three main ad types in their apps to increase their profits.
- When determining how many different ad libraries to implement in an Android app, a developer should avoid including too many. According to the statistics provided by this research, empirically no more than 6 ad Networks should be added into an app to avoid bad ratings.
- Developers should take into account the different stages of the app's life cycles when deciding how many ad networks to implement in their apps. When the app is at its initial stage, developers should use at most one ad network. They should use more when the app becomes more popular.
- The authors propose that developers should opt to mainly use Embedded and Popup ads during any stage of the app's life cycle. Aggressive ad types should be used when their application becomes more mature.
- Even though the previous implications may suggest that developers should stick to Low-Aggressiveness ads, after analyzing the Aggressiveness of the ads hosted by various applications, the research paper concludes a different idea. Developers can use both Low and High Aggressiveness ads at the same time to maximize their revenue. However, this practice should be carefully implemented and thoughtfully monitored.

#### VI. LIMITATIONS AND FUTURE WORK

MAdLens takes the static approach of analyzing and extracting the APIs from Android applications. Therefore, it has the inherited limitations of static analysis. The system counts the number of occurrences of various APIs at a specific time and because of the code control flow, some portions of the code are not necessarily executed. Thus, investigated apps could have had additional ads which were not captured by the system because they were not displayed when the experiment was conducted. There are dynamic approaches which can inspect an app's run-time behavior, but these are still very challenging to implement at a large scale. A viable alternative proposed in the paper was a dynamic solution based on image recognition techniques to identify ad activities on the user interface. However, this option was left by the authors for future work as it required significant additional research.

## VII. CONCLUSION

The analyzed paper represents one major step forward for the in-app advertising industry. It manages to fill in a gap caused by the lack of theoretical knowledge by providing insights for developers about how to optimize their app monetization with optimal ad placement choices. The authors collected 277,616 Android apps belonging to more than 48 categories and performed a comprehensive exploration before coming up with valuable results. These findings are able guide any mobile app programmer into Android in-app ad best practice.

## REFERENCES

- [1] B. He, H. Xu, L. Jin, G. Guo, Y. Chen, and G. Weng, "An investigation into android in-app ad practice: Implications for app developers," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, 2018, pp. 2465–2473.
- [2] Gartner says worldwide sales of smartphones grew 9 percent in first quarter of 2017. [Online]. Available: <https://www.gartner.com/en/newsroom/press-releases/2017-05-23-gartner-says-worldwide-sales-of-smartphones-grew-9-percent-in-first-quarter-of-2017>
- [3] VisionMobile. App economy forecasts 2014 - 2017.
- [4] M. C. Grace, W. Zhou, X. Jiang, and A.-R. Sadeghi, "Unsafe exposure analysis of mobile in-app advertisements," in *Proceedings of the Fifth ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WISEC '12. New York, NY, USA: ACM, 2012, pp. 101–112. [Online]. Available: <http://doi.acm.org/10.1145/2185448.2185464>
- [5] W. Meng, R. Ding, S. P. Chung, S. Han, and W. Lee, "The price of free: Privacy leakage in personalized mobile in-apps ads," in *NDSS*, 2016.
- [6] I. Leontiadis, C. Efstratiou, M. Picone, and C. Mascolo, "Don't kill my ads!: balancing privacy in an ad-supported mobile application market," in *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*. ACM, 2012, p. 2.
- [7] J. Crussell, R. Stevens, and H. Chen, "Madfraud: Investigating ad fraud in android applications," in *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. ACM, 2014, pp. 123–134.
- [8] S. Nath, "Madscope: Characterizing mobile in-app targeted ads," in *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 2015, pp. 59–73.
- [9] B. Liu, B. Liu, H. Jin, and R. Govindan, "Efficient privilege de-escalation for ad libraries in mobile apps," in *Proceedings of the 13th annual international conference on mobile systems, applications, and services*. ACM, 2015, pp. 89–103.