Code-Level Dark Patterns: Exploring Ad Networks' Misleading Code Samples with Negative Consequences for Users

Mohammad Tahaei
mohammad.tahaei@ed.ac.uk
School of Informatics
University of Edinburgh

Kami Vaniea
kami.vaniea@ed.ac.uk
School of Informatics
University of Edinburgh

ABSTRACT

We introduce code-level dark patterns in ad networks. These are official code samples provided by ad networks that will result in user-facing dark patterns, if copy-pasted by developers. Developers who do not carefully read the code for all the nuanced consequences may endanger users privacy by using these code samples. We present three code samples from Google and Amazon ad networks where the code samples do not provide a "I do not consent" option for location data collection, the consent form keeps reappearing until the user consents, and the inclusion of unnecessary permissions when they are presented as "optional" in the surrounding text.

CCS CONCEPTS

• **Security and privacy** → **Human and societal aspects of security and privacy**; • **Software and its engineering** → *Software creation and management*; • **Information systems** → **Online advertising**.

KEYWORDS

software developers, usable privacy, ad networks

ACM Reference Format:
Mohammad Tahaei and Kami Vaniea. 2021. Code-Level Dark Patterns: Exploring Ad Networks' Misleading Code Samples with Negative Consequences for Users. *Position Paper at the "What Can CHI Do About Dark Patterns?" Workshop at CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* 4 pages.

1 MOTIVATION

Ad networks are one of the primary monetisation methods in the mobile app market [2, 3, 6, 9, 11, 12], about 77% of free Android apps contain an ad network [7, 8]. However, developers who choose to monetise their apps using ad networks are not always fully informed about privacy consequences of their decisions on users and tend to follow the default options provided by ad networks [11]. Some of these options are graphical interfaces with radio buttons and checkboxes, and some of them are code samples. In this study, we focus on the code samples as an official source provided by ad networks because developers are known to use code samples on the Internet to build their applications. We are keen to understand the consequences of using these code sample on the users. In the security domain, for example, copy-pasting code from online sources has resulted in severe security consequences such as leaving apps open to man-in-the-middle attacks [4].

We looked at four most popular ad networks on Android [1] which were: *Google AdMob*, *Amazon Mobile Ad Network*, *Facebook Audience Network*, and *Twitter MoPub* to search for any defaults that are not in favour of users and instead favour the ad networks. For the purpose of this paper, we focus on the code-level dark patterns, an extended version of this paper was available at CHI '21 Late-Breaking Work [15].

We define code-level dark patterns as official sample code provided by a large platform (e.g. Amazon, Apple, Facebook, Google, and Twitter) that has a negative consequence for users, leading to a higher data collection and user interfaces with dark patterns. These code samples are deceptive code samples that developers may copy-paste into their apps without reading or knowing about what each line of the code does, and the consequences of their copy-pasting behaviour. Mainly, in this short paper, we focus on three code samples with privacy consequences.

2 DISCOVERED CODE-LEVEL DARK PATTERNS

Google AdMob and Amazon Mobile Ad Network provide code samples and libraries to Android mobile developers to easily integrate their ad networks. We provide three code samples from these two ad networks that if developers copy and paste them into their apps, they all lead to a dark pattern on the user side. All screenshots were taken in Feb '21.

Consent form without a "I do not consent" option. Google AdMob asks developers to adhere to its location data policy and provides a code sample to inform users about their data collection policy (Figure 3). The code sample says "We may use your location" when Google AdMob definitely does use the data for personalisation purposes. The included example privacy policy link also leads to a Chinese Android market website. Finally, the options presented to the user will only have an "OK" button without providing any "I do not consent" option. If a developer were to copy this code from the official Google AdMob's page verbatim, it would compile but their users will not have accurate privacy policy information or the option to not consent. The resulting user-facing dialogue is shown in Figure 2.

Consent form that only disappears with a consent. The second code sample from Google AdMob shows a consent form to the user (Figure 3). However, it will keep appearing on the user's screen until they give consent. The `loadForm()` method in the loop will be called continuously until the user consents.

CHI '21 "What Can CHI Do About Dark Patterns?" Workshop, May 8–13, 2021, Yokohama, Japan

```
protected void presentConsentOverlay(Context context) {
  new AlertDialog.Builder(context)
      .setTitle("Location data")
      .setMessage("We may use your location, " +
          "and share it with third parties, " +
          "for the purposes of personalized advertising, " +
          "analytics, and attribution. " +
          "To learn more, visit our privacy policy " +
          "at https://myapp.com/privacy.")
  .setNeutralButton("OK", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {
      dialog.cancel();
      // TODO: replace the below log statement with code that specifies how
      // you want to handle the user's acknowledgement.
      Log.d("MyApp", "Got consent.");
    }
  })
  .show();
}

// To use the above method:
presentConsentOverlay(this);
```

Figure 1: Precise Location Data Policy page in Google AdMob provide a sample code for obtaining location consent from users without providing a "I do not consent" or a "Reject" button. Developers who use this sample code spread a "forced action" dark pattern in their apps. The provided URL to "our" privacy policy leads to a Chinese Android app market page. The use of "We may use your location" is also misleading because the location data is being used for this purpose.

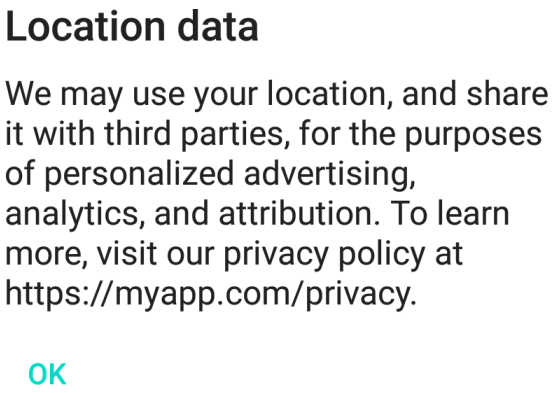

Figure 2: Resulting user-facing dialogue from the code sample in Figure 1. There are no options to cancel or disagree with the policy.

Collecting more data is optional, but it may yield to higher revenue. The third example is from Amazon Mobile Ad Network (Figure 4). Amazon Mobile Ad Network suggests developers provide the ad network with access to fine and coarse location, network, and WiFi state of the user's phone. While these are not essential for the library's functionality, they are highly recommended, and Amazon Mobile Ad Network claims that they may result in higher revenue for the developer: "These additional permissions allow Amazon to provide relevant, targeted ads to your users, which may result in higher CPMs."

3 IMPLICATIONS AND SUGGESTIONS

Non-compliant GDPR consent forms [5, 10, 17], lack of decent privacy policies in apps, and questions developers ask in Stack Overflow about privacy [16] show that privacy is not an easy task for developers [13, 14]; in particular with the recent introduction and changes to regulations and laws. Developers, however, tend to follow requirements posed by platforms such as Apple and Google [16]. When Google defines new sensitive data or permissions, developers must follow the new requirement or their apps may not end up in the app store. The strong influence of platforms on the privacy ecosystem is, therefore, undeniable. In the case of Android, Google

```
public void loadForm(){
    UserMessagingPlatform.loadConsentForm(
        this,
        new UserMessagingPlatform.OnConsentFormLoadSuccessListener() {
            @Override
            public void onConsentFormLoadSuccess(ConsentForm consentForm) {
                MainActivity.this.consentForm = consentForm;
                if(consentInformation.getConsentStatus() == ConsentInformation.ConsentStatus.REQUIRED) {
                    consentForm.show(
                        MainActivity.this,
                            new ConsentForm.OnConsentFormDismissedListener() {
                                @Override
                                public void onConsentFormDismissed(@Nullable FormError formError) {
                                    // Handle dismissal by reloading form.
                                    loadForm();
                                }
                            });

                }

            }
        },
        new UserMessagingPlatform.OnConsentFormLoadFailureListener() {
            @Override
            public void onConsentFormLoadFailure(FormError formError) {
                /// Handle Error.
            }
        }
    );
}
```

Figure 3: Obtaining Consent with the User Messaging Platform page in Google AdMob provides a sample code for obtaining consent from users that constantly shows the form to the user until they consent. Developers who use this sample code spread a "nagging" dark pattern in their apps.

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
```

Figure 4: Quick Start Guide page in Amazon Mobile Ad Network provides a sample code for the manifest file for Android developers which includes permissions to fine and coarse locations, network state, and WiFi state. While these are not required for the API to work, they are "highly recommended" and "allow Amazon to provide relevant, targeted ads to your users, which may result in higher CPMs."

Play, and Google AdMob, they are all related to one company, expecting a fair assessment of what should be private and what should not be considered private is difficult. Google Play can stop developers from propagating dark patterns to users by introducing new requirements or running automatic checks before rolling out an app on Google Play. However, Google AdMob (with a conflict of interest with Google Play) may be one of the sources of these dark patterns, expecting Google Play to operate fair without any prejudices is not easy. Disentangling these actors may break the cycle and be one potential solution; having actors who would work towards different goals instead of working for the same goal and corporation. While this can be a long term solution, in the meantime as a short time solution, consent forms can be built by trusted open-source third-parties, with usable interfaces for developers.

ACKNOWLEDGMENTS

This work was sponsored in part by Microsoft Research through its PhD Scholarship Program and a Google Research Award.

REFERENCES

[1] Md Ahasanuzzaman, Safwat Hassan, Cor-Paul Bezemer, and Ahmed E. Hassan. 2020. A longitudinal study of popular ad libraries in the Google Play Store. *Empirical Software Engineering* 25, 1 (Jan. 2020), 824–858. https://doi.org/10.1007/s10664-019-09766-x

[2] App Annie. 2020. The State of Mobile in 2020. Retrieved August 2020 from https://www.appannie.com/en/insights/market-data/state-of-mobile-2020/

[3] AppBrain. 2020. Android Ad Network statistics and market share. Retrieved August 2020 from https://www.appbrain.com/stats/libraries/ad-networks

[4] Felix Fischer, Konstantin Böttinger, Huang Xiao, Christian Stransky, Yasemin Acar, Michael Backes, and Sascha Fahl. 2017. Stack Overflow Considered Harmful? The Impact of Copy Paste on Android Application Security. In *2017 IEEE Symposium on Security and Privacy (SP)*. 121–136. https://doi.org/10.1109/SP.2017.31

[5] Imane Fouad, Cristiana Santos, Feras Al Kassar, Nataliia Bielova, and Stefano Calzavara. 2020. On Compliance of Cookie Purposes with the Purpose Specification Principle. In *IWPE 2020 - International Workshop on Privacy Engineering*. 1–8. https://hal.inria.fr/hal-02567022

[6] Catherine Han, Irwin Reyes, Álvaro Feal, Joel Reardon, Primal Wijesekera, Amit Elazari, Kenneth A Bamberger, and Serge Egelman. 2020. The Price is (Not) Right: Comparing Privacy in Free and Paid Apps. In *Privacy Enhancing Technologies Symposium (PETS 2020)*. 21. https://doi.org/10.2478/popets-2020-0050

[7] Boyuan He, Haitao Xu, Ling Jin, Guanyu Guo, Yan Chen, and Guangyao Weng. 2018. An Investigation into Android In-App Ad Practice: Implications for App Developers. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*. 2465–2473. https://doi.org/10.1109/INFOCOM.2018.8486010

[8] Ling Jin, Boyuan He, Guangyao Weng, Haitao Xu, Yan Chen, and Guanyu Guo. 2021. MAdLens: Investigating into Android In-App Ad Practice at API Granularity. *IEEE Transactions on Mobile Computing* 20, 3 (2021), 1138–1155. https://doi.org/10.1109/TMC.2019.2953609

[9] Ilias Leontiadis, Christos Efstratiou, Marco Picone, and Cecilia Mascolo. 2012. 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* (San Diego, California) *(HotMobile '12)*. Article 2, 6 pages. https://doi.org/10.1145/2162081.2162084

[10] Celestin Matte, Nataliia Bielova, and Cristiana Santos. 2020. Do Cookie Banners Respect my Choice? : Measuring Legal Compliance of Banners from IAB Europe's Transparency and Consent Framework. In *2020 IEEE Symposium on Security and Privacy (SP)*. 791–809. https://doi.org/10.1109/SP40000.2020.00076

[11] Abraham H. Mhaidli, Yixin Zou, and Florian Schaub. 2019. "We Can't Live Without Them!" App Developers' Adoption of Ad Networks and Their Considerations of Consumer Risks. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. 20. https://www.usenix.org/conference/soups2019/presentation/mhaidli

[12] Statista. 2020. Share of global smartphone shipments by operating system from 2014 to 2023. Retrieved August 2020 from https://www.statista.com/statistics/272307/market-share-forecast-for-smartphone-operating-systems/

[13] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. 1–15. https://doi.org/10.1145/3411764.3445768

[14] Mohammad Tahaei and Kami Vaniea. 2019. A Survey on Developer-Centred Security. In *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. 129–138. https://doi.org/10.1109/EuroSPW.2019.00021

[15] Mohammad Tahaei and Kami Vaniea. 2021. "Developers Are Responsible": What Ad Networks Tell Developers About Privacy. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '21 Extended Abstracts)*. 1–12. https://doi.org/10.1145/3411763.3451805

[16] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. 2020. Understanding Privacy-Related Questions on Stack Overflow. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. 1–14. https://doi.org/10.1145/3313831.3376768

[17] Christine Utz, Martin Degeling, Sascha Fahl, Florian Schaub, and Thorsten Holz. 2019. (Un)Informed Consent: Studying GDPR Consent Notices in the Field. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (London, United Kingdom) *(CCS '19)*. 973–990. https://doi.org/10.1145/3319535.3354212