Pièce 3 / Exhibit 3

Methodology and technical specifications of the mobile application testing environment

Introduction

In 2023, *noyb* launched a project to investigate the mobile applications (hereinafter "mobile apps"; "apps") environment.

In May 2023, *noyb* analysed three mobile apps installed on an Android user's mobile phone.

The purpose of the analysis was to collect insights about the data sharing between mobile apps and third-party tracking companies. The project showed that all investigated apps shared user's personal data to third-party tracking companies illegally.

The current report serves as an attachment to the GDPR complaints which were filed with the French supervisory authority, the CNIL, as a result of the identified illegal data sharing.

Description of the project

On [redacted] 2023, a French user (hereinafter "Complainant") installed three mobile apps on their Android mobile device.

1. The goal

The objective of the project was to investigate the extent to which the installed apps complied with the provisions of the ePrivacy Directive and the General Data Protection Regulation (GDPR), while also examining how the apps handled users' personal data.

Previous research revealed[1] that mobile apps share a large amount of personal data with third-party tracking companies by way of integration of Software Development Kits (SDKs). Such data sharing is in most cases unknown to mobile apps users and happens even before a user consents to the data sharing intended for e.g. personalised advertising, extensive user analytics or profiling.

noyb's project is designed to analyse the status quo of the data sharing between mobile apps and third-party tracking companies (which provide SDK integration for the apps) and take enforcement action in cases where violations are identified. The current document serves as a description of the methodology applied during the first stage of the project.

[1] Konrad Kollnig et. al., *A Fait Accompli? An Empirical Study into the Absence of Consent to Third-Party Tracking in Android Apps*: https://www.usenix.org/system/files/soups2021-kollnig.pdf (accessed on April 13, 2023); Konrad Kollnig et. al., *Before and after GDPR: tracking in mobile apps*: https://policyreview.info/articles/analysis/and-after-gdpr-tracking-mobile-apps (accessed on April 13, 2023); Trung Tin Nguyen et. al., *Share First, Ask Later (or Never?) Studying Violations of GDPR's Explicit Consent in Android Apps*: https://www.usenix.org/system/files/sec21-nguyen.pdf (accessed on April 13, 2023); Benjamin Altpeter, *Worrying confessions: A look at data safety labels on Android*: https://www.datarequests.org/blog/android-data-safety-labels-analysis/

Pièce 3 / Exhibit 3

2. App selection criteria

The apps were selected based on:

- The country of the mobile app developer's main establishment (France and the United States);
- the number of downloads according to Google Play Store (>1 million downloads); and
- the presence of known analytics and advertising trackers using the reports provided by Exodus Privacy.

The list of all apk ("Android Package Kit") versions of the apps analysed is attached in separate documents (Annex [redacted] "Fnac", Annex [redacted] "SeLoger", Annex [redacted] "MyFitnessPal").

3. Tools

To achieve the set goal, the project relied on the PiRogue Tool Suite provided by the Defensive Lab Agency. The PiRogue tool is designed to capture, detect and identify privacy violations in Android applications. The tool is built on top of the Android Debug Bridge (ADB) and Frida. ADB is a command line tool that allows developers to communicate with an Android device.

PiRogue uses Frida to retrieve TLS encryption keys, socket activity, device screen recording and RSA/AES operations by instrumenting the operating system without modifying the behaviour of the target application. Unlike tools such as MITM-proxy, PTS allows TLS traffic decryption without tampering the network traffic or introducing any application-level security vulnerability such as disabling certificate pinning.

The primary approach used to analyse the apps was traffic analysis. Traffic analysis involves capturing the data transmitted from mobile applications and analysing the data transmissions for any privacy violations.

The implementation of security good practices at the application level implies that most of the traffic is encrypted with TLS. With the use of the PiRogue[2] toolkit it was possible to extract encryption keys from the Complainant's device memory and to capture the entire network traffic that occurred during the run of the analysed mobile apps. The network traffic can be subsequently decrypted with standard tools such as Wireshark by providing both the PCAP file and the SSL keylog file.

For convenience, we analysed the network traffic using another tool from the PiRogue Tool Suite, called "Colander" (the tool is not publically available yet). Colander enables a user to quickly identify any suspicious network connections or data sharing practices by applying detection rules directly to the decrypted traffic. Colander can also recover data that has been encrypted before being transmitted by looking at the cryptographic operations that occurred during the execution of the target application. An analysis report is automatically issued listing:

[2] See package versions in Annex 4.

Pièce 3 / Exhibit 3

- All analysis artifacts such as PCAP file containing traffic data or screen recording and their digital signatures;
- the communication direction (inbound or outbound);
- the source and destination host, IP address and organisation;
- the technical part of the application that has been handling the given data transmission (internal application code or 3rd-party SDKs);
- the identification and classification of the transmitted data such as advertising id, location data etc.;
- when applicable, the inferred purpose (such as analytics or advertisement) of the data collection using Exodus Privacy tracker classification to identify the recipient company.

Colander analysis and detection accuracy can be assessed by opening analysis artifacts with Wireshark and manually verifying each data transmission.

The Colander documentation generated for the analysed mobile apps is attached separately (Annex [redacted] "Fnac Colander documentation", Annex [redacted] "SeLoger Colander documentation", Annex [redacted] "MyFitnessPal Colander documentation").

The entire source code of the PiRogue Tool Suite can be audited at https://github.com/PiRogueToolSuite/.

4. Device description

The mobile apps were installed and run on a rooted[3] Android device with the following specifications:

- Device model: Samsung S9+ [redacted]
- Android version: 10 [redacted]
- Operating system (OS): /e/ Murena OS;
- OS image: [redacted]
- Recovery image: [redacted]
- Serial number: [redacted]
- IMEI: [redacted]
- Network operator: [redacted]
- SSID: [redacted]

The modified OS used on the mobile device creates a unique privacy enhanced environment[4] with the following features:

- Any feature or code that enables the transmission of data to Google servers disabled or at the very least the accesses were anonymised;
- Google Services replaced by microG and by alternative services;

[3] The rooting of the device was conducted according to the methodology described in section 5 below.

[4] See more details here: https://e.foundation/e-os/

- all Google apps removed and replaced by equivalent Open Source applications;
- no use of Google servers to check connectivity;
- NTP servers that do not belong to Google used;
- geolocation using Mozilla Location Services in addition to GPS.

Since a standard Android phone does not come with the specific privacy features provided by /e/ Murena OS, it was decided for the purposes of this project to disable the "Advanced Privacy" /e/ Murena OS-specific feature turning off:

- The filtering of trackers network traffic (blocking the network traffic);
- the mocking of the geolocation (faking the real location);
- the mocking of the real IP address of the device (faking the IP address).

Advertising IDs

As mentioned in the previous section, the version of the Android OS on the Complainant's device used an alternative to Google Play Services called "MicroG"[5]. MicroG never returns the same Android advertising identifier (hereinafter "AdID"). Every time a mobile app reads it, a new AdID is generated and returned. In other words, if a given mobile application reads the AdID three times, the OS will return three different advertising identifiers.

The OS returned the following AdIDs[6] for the three analysed mobile apps:

- [redacted]

5. Preparation of the device

The Complainant's device was rooted by flashing the OS image listed in Section 4 with the program "*easy-installer*" provided by *e* foundation (https://doc.e.foundation/easy-installer). The OS image, also provided by *e* foundation, is rooted by default.

6. Test description

Each app was installed on the Complainant's device one at a time, launched and run for a duration of ca. 1-2 minutes without any user interaction with either the app or (any) pop-up banner.

The Complainant was not logged into any of the apps.

The Complainant manually shut down each app.

A recording of the mobile app launch, run and shutdown is available in the documentation for each case.

[5] https://microg.org/

[6] The command 'pirogue-intercept-gated' command creates a file containing all issued AdIDs. Its content is then issued in the detection rules. It is possible to check the log in the documentation of each case.

Pièce 3 / Exhibit 3

The device was connected to the Pirogue setup through Wi-Fi and USB. This connection enabled the monitoring of the network traffic on the phone.

Internet access was provided through [redacted] ISP.

7. Results

The analysis revealed that all the three mobile apps contacted a number of third-party tracking services immediately after launching.

The apps illegally shared the following personal data with third-party tracking companies:

- Google Advertising ID (AdID);
- device model;
- device brand;
- local IP address;
- mobile operator name;
- unique user IDs (UUID);
- some fingerprinting data (eg screen width, screen height, battery status, language etc.).

In some cases, additional user identifiers were generated on the Complainant's device.

8. Conclusions

The study provided valuable insights into the data sharing practices of mobile apps revealing that all the analysed apps start sharing the individual's data as soon as the apps are launched. The observed data sharing is illegal because it happens without the individual's consent.

PiRogue and Colander tools (PiRogue Tool Suite) developed by the Defensive Lab Agency were instrumental in conducting the analysis and identifying potential privacy violations in the analysed apps.

The current setup had its limitations (for example, it is not possible to decrypt the network traffic generated by the operating system), and there may be data transmissions and interactions with third parties that the study was not able to observe directly due to the use of a highly privacy respecting set up of the OS and its features.

The current project will be expanded to analyse a higher number of mobile apps and SDKs, potentially on a different version of the Android OS.