USENIX'23 Artifact Appendix: Security and Privacy Failures in Popular 2FA Apps

Conor Gilsenan
UC Berkeley / ICSI

Fuzail Shakir
UC Berkeley

Noura Alomar
UC Berkeley

Serge Egelman
UC Berkeley / ICSI

A Artifact Appendix

A.1 Abstract

The repo on GitHub contains the following artifacts:

Search Terms The list of search terms that we used to identify as many TOTP apps in the Google Play Store as possible (see Section 4.1 - App Selection);

App Checklists For each app, the customized checklist that enumerates exactly which actions to take within the app and which data to enter while recording the network traffic (see Section 4.2.1 - Exploring the App); and

Decryption Scripts For each app that supports encrypted TOTP backups, the golang script that implements the decryption process (see Section 4.2.3 - Performing Cryptanalysis).

Along with the instructions in the README, the app checklists and decryption scripts allow researchers to reproduce the findings we report in Tables 1, 2, and 3 in the paper.

A.2 Description & Requirements

A.2.1 Security, privacy, and ethical concerns

The README instructions included in the repo suggest flashing a new Android ROM onto the device in order to use root privileges to capture the plaintext traffic generated by each TOTP app. Rooting a phone can have negative security consequences and flashing a new Android ROM will entirely wipe the phone's data, so we do not recommend using your primary Android device. If available, use an old Android phone that you can flash with a stock Android ROM after evaluation.

The backup mechanisms on several TOTP apps require divulging personal information, such as email address, phone number, name, and date of birth. To protect your privacy, we recommend using fake values where possible. Create a new email address specifically for the purpose of evaluation. Many apps require an active phone number that can receive SMS messages for authentication purposes. During our work, we purchased temporary phone numbers from messagebird.com to protect our privacy. Other telephony APIs available online can achieve the same privacy goals.

A.2.2 How to access

Publicly available at https://github.com/blues-lab/totp-app-analysis-public.

The following tag is intended for review by the USENIX 2023 Artifact Evaluation committee: https://github.com/blues-lab/totp-app-analysis-public/releases/tag/usenix-sec23-ae.

A.2.3 Hardware dependencies

During our research, we used Pixel 3a Android phones running a custom version of Android 9. However, our results can be replicated using any Android phone running version 9+.

A.2.4 Software dependencies

The README contains detailed instructions on how to install many of the following software dependencies:

- The decryption scripts require golang v1.18 or higher. The golang website provides installation documentation: https://tip.golang.org/doc/install
- The Android Debug Bridge (adb) is required to control the Android phone from the researcher's machine: https://developer.android.com/studio/command-line/adb
- Magisk is a suite of open source software for customizing Android: https://github.com/topjohnwu/Magisk/blob/master/docs/install.md
- mitmproxy is a free and open source interactive HTTPS proxy: https://mitmproxy.org/
- (Optional) The scrcpy tool allows the researcher to mirror the Android phone's screen onto their computer and, optionally, record the phone's screen: https://github.com/Genymobile/scrcpy

A.2.5 Benchmarks

None

A.3 Set-up

Please see the README in the artifact repo on GitHub.

A.3.1 Installation

Please see the README in the artifact repo on GitHub.

A.3.2 Basic Test

After following the instructions in the repo's README, you should be able to capture plaintext traffic generated by each TOTP app that you are evaluating.

A.4 Evaluation workflow

The README in the linked repo contains detailed steps to reproduce our findings for each TOTP app we analyzed.

Once recorded, the plaintext traffic can be analyzed to confirm whether TOTP fields (secret, issuer, label) are sent in plaintext. Additionally, values from the plaintext can be copy/pasted into the corresponding decryption scripts to verify the cryptographic primitives used in each TOTP backup mechanism.

A.4.1 Major Claims

Our major claims are enumerated in the following tables in the paper:

Table 1: Overview of the backup mechanisms supported in each app.
Table 2: Overview of the backup mechanisms that automatically sync data to the cloud.
Table 3: Cryptographic details of app backup mechanisms.

Evaluators should be able to verify and reproduce the findings reported in each cell of Tables 1, 2, and 3.

A.4.2 Experiments

The linked repo contains a custom checklist for each TOTP app, which enumerates exactly which actions to take within the app and which data to enter while recording the network traffic (see Section 4.2.1 - Exploring the App). It should take about 10-20 minutes to execute the steps defined in the checklist for each app.

A.5 Version

Based on the LaTeX template for Artifact Evaluation V20220926. Submission, reviewing and badging methodology followed for the evaluation of this artifact can be found at https://secartifacts.github.io/usenixsec2023/.