Response to "Security and Privacy Failures in Popular 2FA Apps" Report [1]
Date: November 19, 2024

We would like to express our appreciation to Conor Gilsenan, Fuzail Shakir, Noura Alomar, and Serge Egelman, along with their respective affiliations at UC Berkeley and ICSI.

Following a careful review of the points raised, we address specific concerns in detail below.

1. **Section 5.1: Backup without Network**

The report refers to the process of backup through scanning a QR code in Google Authenticator. We question whether this method qualifies as a true "backup." Google Authenticator now refers to this process as "Transfer Accounts," which better aligns with its functionality.

2. **Section 5.3: Remote Backups with Encryption**

The report correctly notes that earlier versions of our app defaulted to storing backups on Google Drive without a password, requiring users to manually enable password encryption after the backup was created without encryption. This process was changed in late 2023, ensuring that users are now prompted to create a password for their backup before it is stored on Google Drive.
Importantly, backups have always been stored in a secure, isolated space on Google Drive, inaccessible to other apps or from the Google Drive web interface. Google introduced a similar approach for Google Authenticator in April 2023, but without implementing any password protection, considering the secure storage to be sufficient.

3. **Section 5.3.1: Password Policies**

The report highlights the need to enforce stronger password requirements. Initially, we provided users with the flexibility to choose their password length, considering the encrypted nature of data transmission, storage in a secure Google Drive space, and the general security of the Android ecosystem (which stores sensitive information). This approach was confirmed by Google, which does not offer the option for password protected backups for their Google Authenticator app.
However, after careful consideration, we agree that enforcing stronger password requirements is a necessary improvement, and we will implement this feature moving forward.

4. **Section 5.3.4: How TOTP Backups are Encrypted**

The report incorrectly identifies the encryption method in the main text, stating that we use AES-CBC with a random IV. However, Table 3 correctly lists the method as AES-GCM. We have always used AES-GCM, which ensures both encryption and data integrity. It appears there was a discrepancy between the written description and the information presented in the table.

5. **Section 5.5: Privacy Implications - PII to Enable Backups**

The report highlights the use of PII for enabling cloud backups, referencing 2FAS in Table 2 as an app accessing name, and photo for this functionality. This is incorrect. 2FAS only uses requestEmail() during Google Sign-In, which grants access solely to the user's email address. Other fields such as name and photo are not accessed

[1] Report available at: https://www.usenix.org/system/files/usenixsecurity23-gilsenan.pdf

because the app does not use the requestProfile() method, which would be required to retrieve such data. These fields remain completely inaccessible and are not processed by the app.

It is worth clarifying that the email address retrieved during Google Sign-In is accessed only by the 2FAS app on the user's device to display locally on the 2FAS Backup screen, helping the user confirm the linked account. This data never leaves the device or reaches 2FAS as developers. This explanation aims to clarify potential misunderstandings from Table 2.

6. **Section 6.2: Recommendations**
   a) **Use of Argon2**: We agree that using Argon2 will enhance backup security against attacks and are actively working on implementing Argon2id in our application.
   b) **30-Second Timer for Import/Export**:
   Our export/import functions are designed to provide immediate feedback, in line with usability principles such as Nielsen's heuristics (1: Visibility of System Status, 3: User Control and Freedom, and 8: Aesthetic and Minimalist Design). While introducing a delay could enhance security in certain attack scenarios, it may also lead to user frustration, especially if they need to wait 30 seconds after entering an incorrect password. This could negatively affect the overall user experience.

   We believe that immediate feedback is essential, particularly during critical actions like data recovery. While security is always a priority, we strive to balance it with user convenience. Nonetheless, we are open to further discussion on how to address these concerns.

We appreciate the thorough work by the authors and look forward to continuously improving both the security and usability of our application.

Best regards,

Marek Bardzinski
CEO at 2FAS.com