Security analysis of Mi Band 4 shows that it a fitness tracker with good security features.

Problem Statement and Goals

Fitness tracker has become an important wearable device. However, since it tracks sensitive information and is always connected to the smartphone of the user, it poses a potential security risk.

The present research which focuses on the security of the Mi Band 4, a wearable fitness tracker by Xiaomi, tries to identify its general architecture and possible security issues. This research tries to analyse three different aspects of security:

- Privacy
- On-Device
- In-Transit

Approach

Understand the general architecture implemented and for each of the aspects of the security analysis, answer the following questions:

- Privacy
  - What information does the Mi Fit app store?
  - Which format is the data stored in?
  - What information is sent to the Xiaomi server?
  - Does any third party have access to this data?

- On-Device
  - How does the firmware look?
  - Can we upload custom malicious resources?
  - Can we exfiltrate data from the device?

- In-Transit
  - How does the device communicate?
  - Is the data encrypted during transit?
  - How the pairing and authentication works?

Results

- Privacy
  - The app stores device data, profile data, activity data and analytics data.
  - Data is stored in both plain text and encrypted format in SQLite database in app’s private storage, thus not exploitable by other apps. Data is sent to Xiaomi server.
  - Uses reputed ad services like AdMob and MoPub and no suspicious communication with any third party is observed.

- On-Device
  - Tracker stores up to 16 MB of user information
  - No way to dump data from tracker. Only way to get information through BLE
  - Firmware protected by CRC32 and xiaomis signature. Users can customize images and “watch faces” through other files

- In-Transit
  - Services require authentication of the device (expect services related to device information and the authentication itself)
  - Data is sent in clear text, eavesdropper could listen and steal data
  - Pairing / Authentication protocol was reversed. Xiaomi’s improved protocol by adding the validation and signature of their servers

Future Work:
Attempt to exploit / look for vulnerabilities in the .res / .bin files that packages .png files and identify internal key (maybe in firmware) that is used to validate the signature of the server.

Pragati Mishra, Arpit Gupta, Santiago Carvajal
Carnegie Mellon University

www.rsaconference.com/securitiescholar

Carnegie Mellon University