1. Used **ADB Shell** to access the app’s internal storage:

***adb shell***

***su***

***cd /data/data/com.sl.SLBiljetter/shared\_prefs/***

***ls***

***cat release\_remote\_config.xml***

1. Navigated to the app’s **Shared Preferences** directory and found

A black screen with white text

AI-generated content may be incorrect.

1. Found **API keys and tokens** inside the release\_remote\_config.XML.

A screen shot of a computer screen

AI-generated content may be incorrect.

1. Extracted **sensitive JSON data** from release\_remote\_config.xml (contained **API keys, secrets, tokens**). And The JSON contains multiple API keys and secrets, which, if valid, could allow unauthorized access to SL services.
2. If "psa\_auth" is an authentication key used for user authorization, an attacker could use it to gain unauthorized access to services.

A screen shot of a computer

AI-generated content may be incorrect.

1. Then **Validated JSON using Python script**

A screenshot of a computer

AI-generated content may be incorrect.

1. **Then testing API keys and secrets** by sending requests to the **SL APIs**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

--A**ccessed the /databases folder** of the app then found something interesting in Base64 format

A black screen with white letters

AI-generated content may be incorrect.

* And tried debugging it.
* A screenshot of a computer

  AI-generated content may be incorrect.
* After thoroughly analyzing the SQLite database stored in the /no\_backup/ directory of the Android app, I did not find any direct security vulnerabilities.
* The database structure, including tables like WorkSpec, WorkProgress, and WorkTag, appears to be functioning normally.
* Additionally, since the data is stored in an app-private directory (/data/data/com.sl.SLBiljetter/), it is already protected from unauthorized access unless the device is rooted.
* However, while there are no immediate threats, best practices such as avoiding data exports to /sdcard/, ensuring proper backup before modifications.

A screenshot of a computer

AI-generated content may be incorrect.

-During analysis,I found that the Jetpack DataStore (\*.preferences\_pb) files contain sensitive information, including authentication tokens (userToken, sertoken\_readable), --JWT tokens, UUIDs, and tracking-related data.

These tokens, if exposed, could potentially be used for unauthorized access, account hijacking, or user tracking.

- The fact that they are **stored in plaintext** without encryption poses a serious security risk, especially on rooted devices where an attacker could extract and misuse them