TP : Pentesting d'AndroGoat (Application Android Vulnérable)

Objectif :

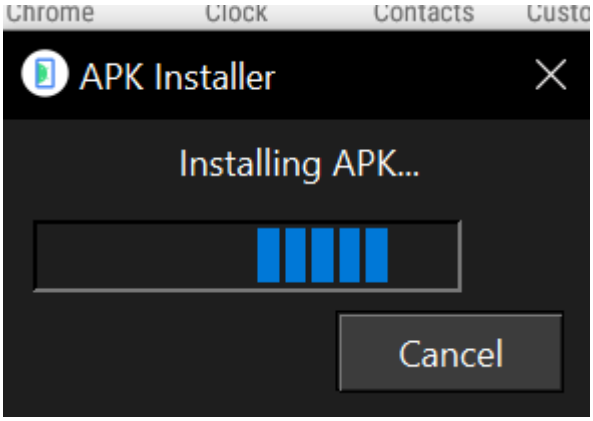
Identifier et exploiter des vulnérabilités dans AndroGoat en utilisant des outils courants de pentesting mobile.

Prérequis :

- Un appareil Android (émulateur ou physique, rooté/débuggable de préférence).
- ADB (Android Debug Bridge).
- Un décompilateur (Jadx-GUI recommandé).
- Un proxy d'interception (Burp Suite Community).
- Application cible : Choisir une application vulnérable (AndroGoat).

1. Configuration du Lab

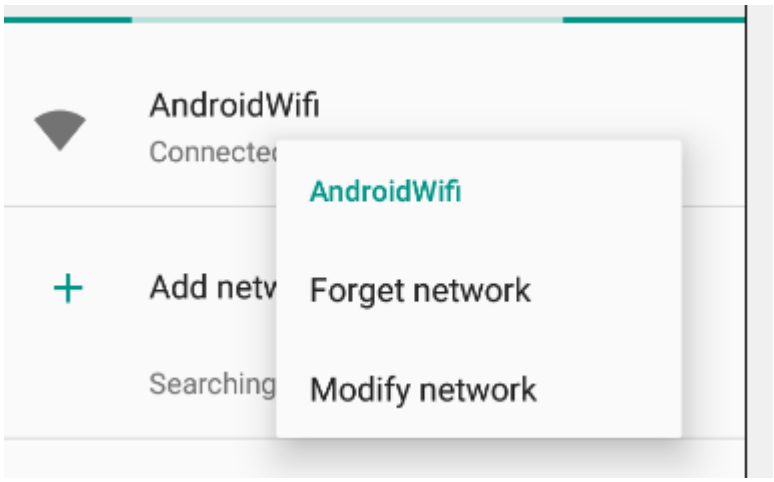
- Étapes :
  - Téléchargez AndroGoat depuis GitHub.
  - Installez l'APK sur un émulateur (Android Studio).

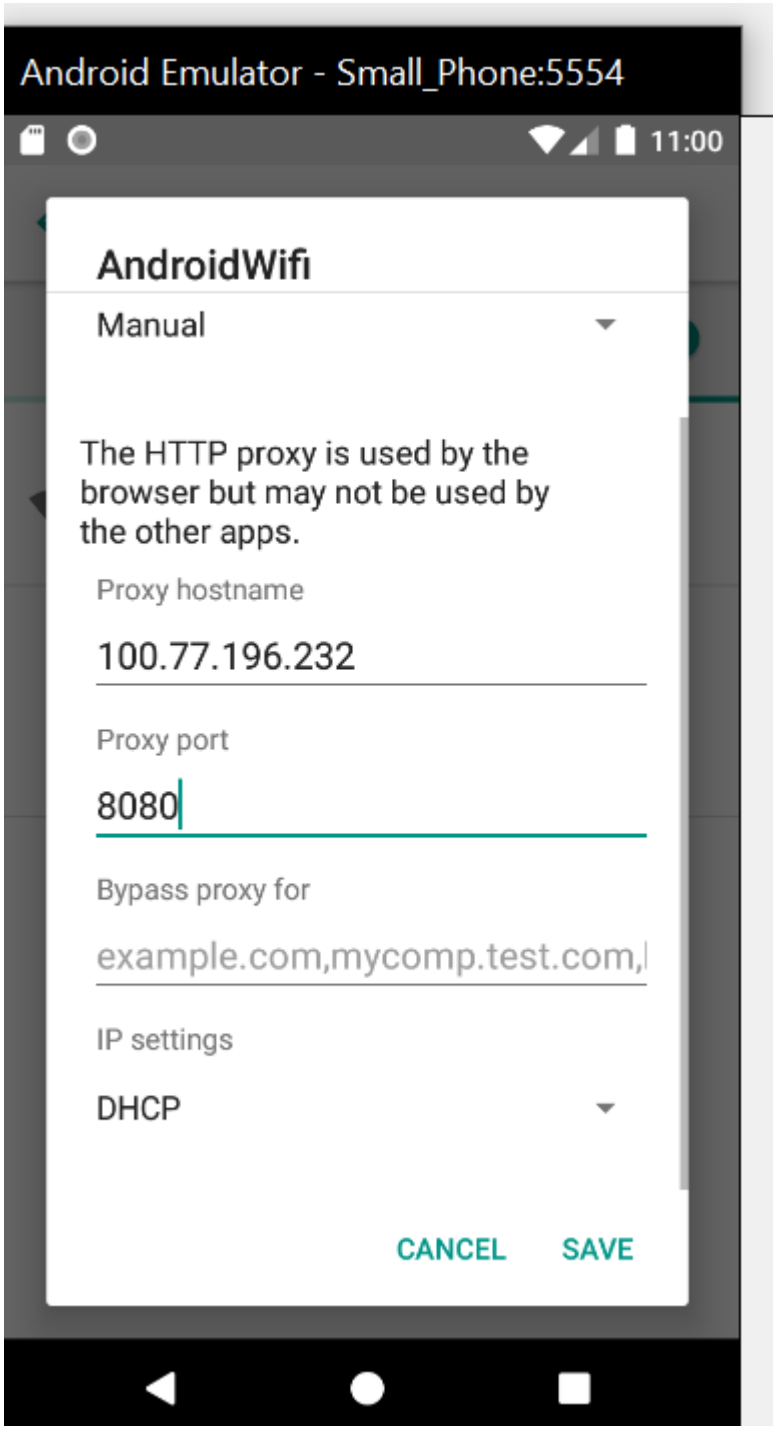


  - Configurez ADB (Android Debug Bridge) pour interagir avec l'appareil :
    - adb devices # Vérifier la connexion
    - adb install AndroGoat.apk

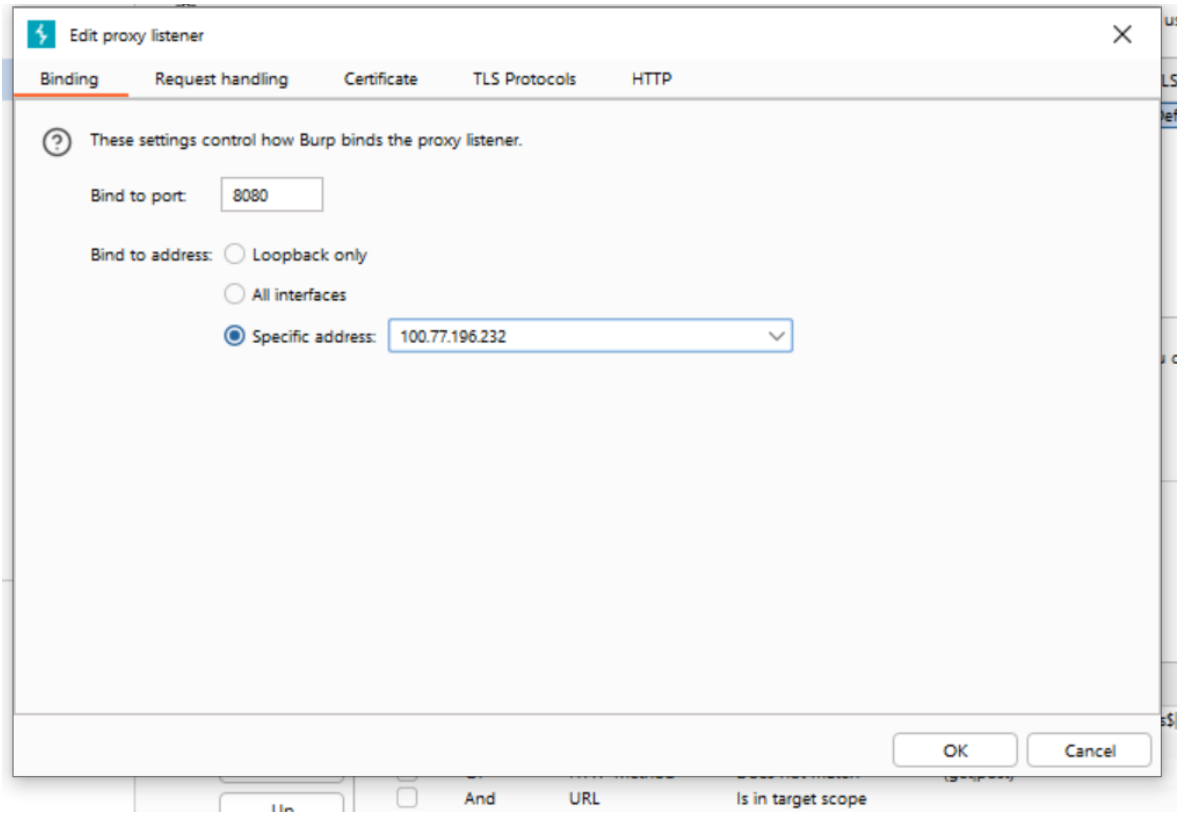
2. Interception du Trafic Réseau

- Configuration d'un Proxy

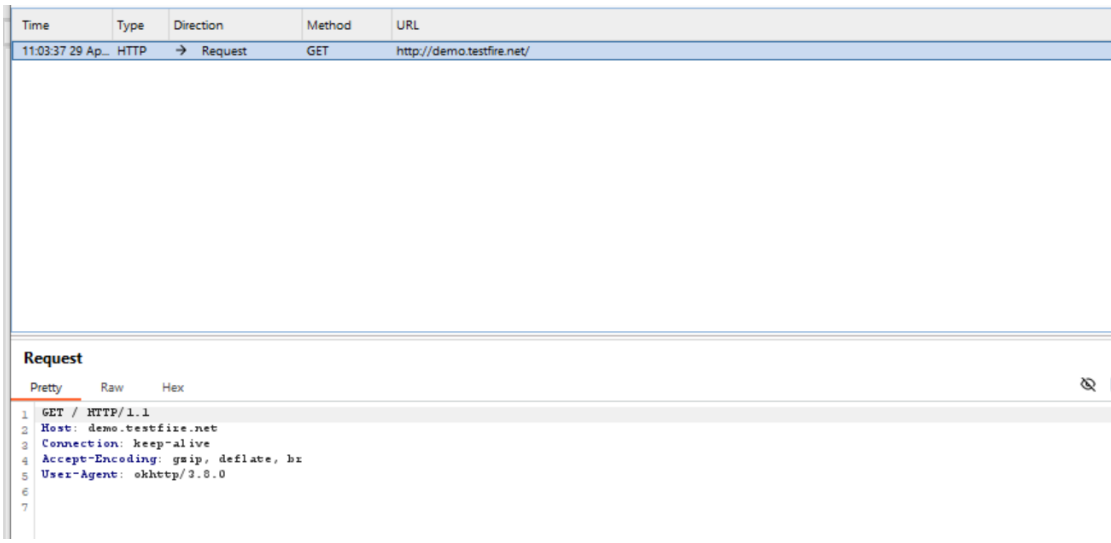

AndroidWifi
Connecte
AndroidWifi
Add netv
Forget network
Searching
Modify network


Android Emulator - Small_Phone:5554
11:00
AndroidWifi
Manual
The HTTP proxy is used by the browser but may not be used by the other apps.
Proxy hostname
100.77.196.232
Proxy port
8080
Bypass proxy for
example.com,mycomp.test.com,
IP settings
DHCP
CANCEL
SAVE

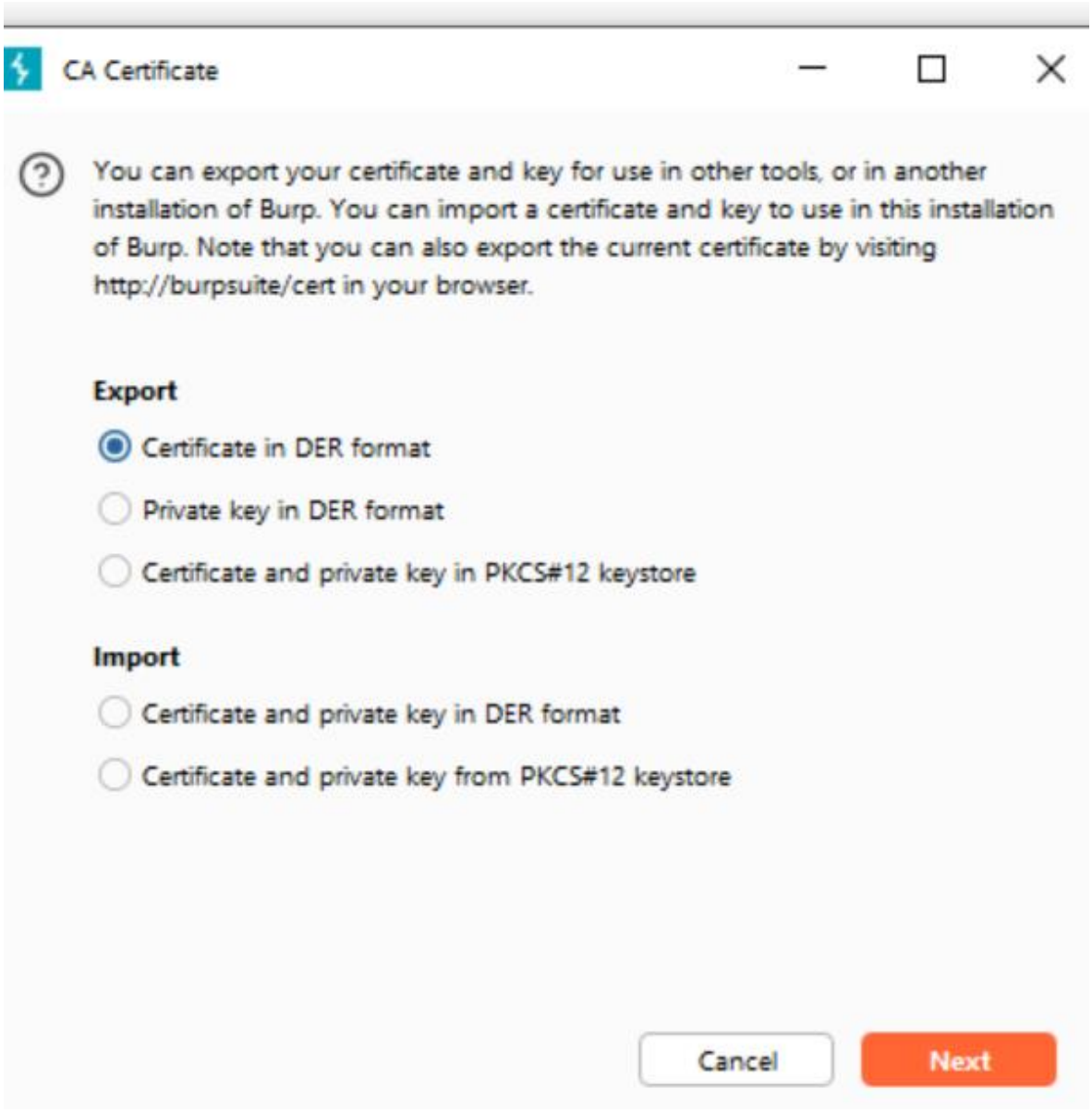
- Définir l'adresse IP de votre machine comme proxy (ex: `192.168.18.231:8080`).

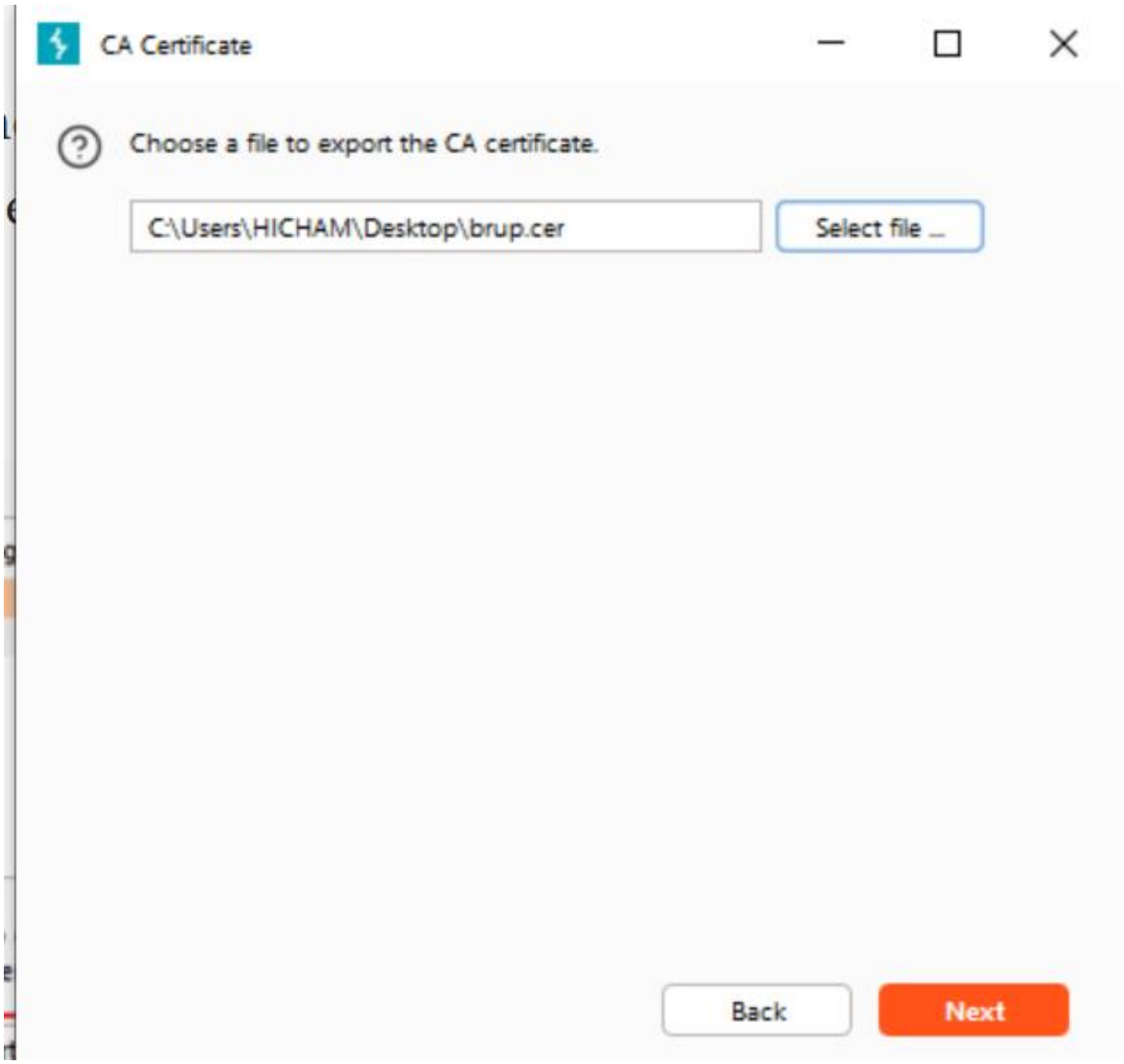


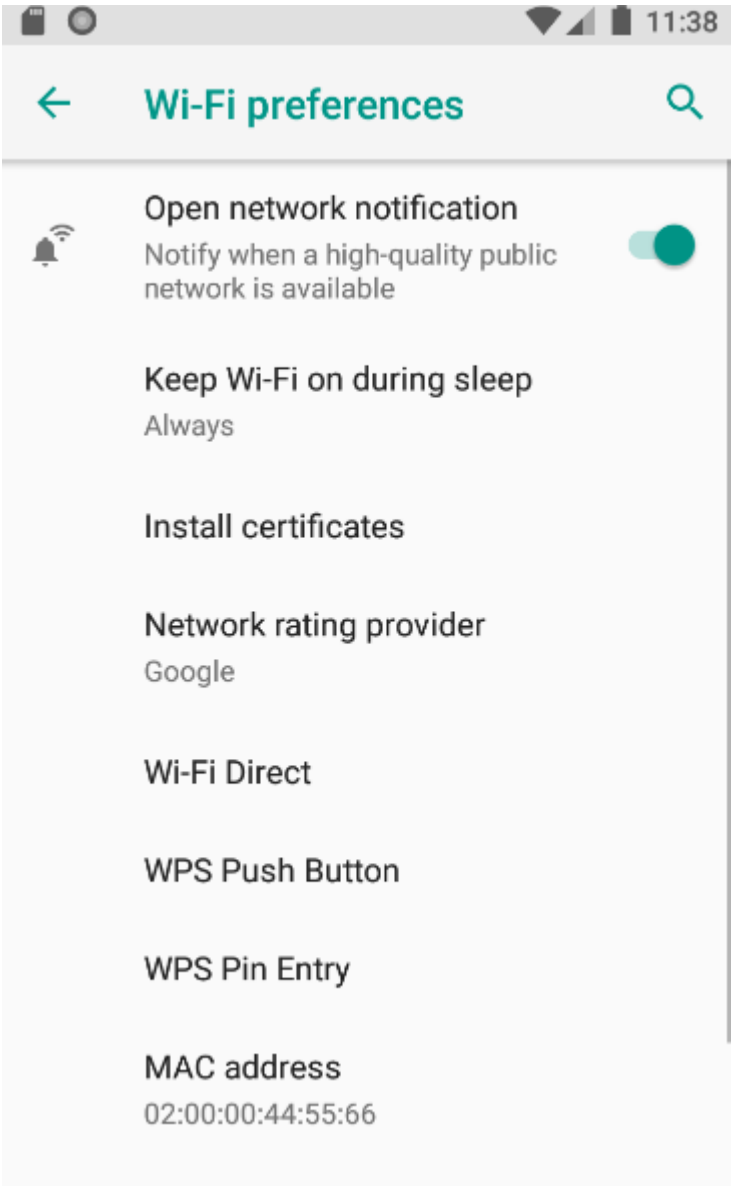
- Dans Burp Suite, configurez un **Proxy Listener** correspondant.
  - Ouvrez AndroGoat et cliquez sur "HTTP" : le trafic sera intercepté dans Burp Suite.

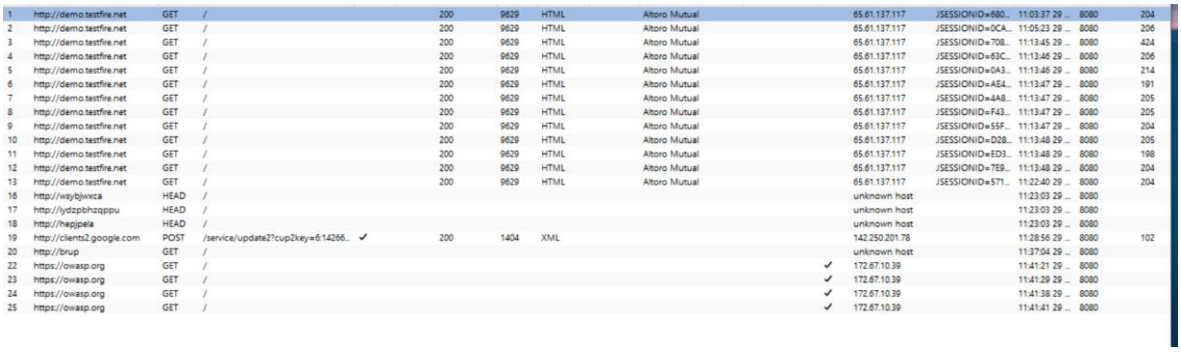


- Pour HTTPS, exportez le certificat Burp (format .cer) et installez-le sur le mobile
    - Accédez aux paramètres Wi-Fi > Certificats > Installez le fichier.
    - Interceptez du trafic HTTPS.


CA Certificate
You can export your certificate and key for use in other tools, or in another installation of Burp. You can import a certificate and key to use in this installation of Burp. Note that you can also export the current certificate by visiting http://burpsuite/cert in your browser.
Export
Certificate in DER format
Private key in DER format
Certificate and private key in PKCS#12 keystore
Import
Certificate and private key in DER format
Certificate and private key from PKCS#12 keystore
Cancel
Next


CA Certificate
Choose a file to export the CA certificate.
C:\Users\HICHAM\Desktop\brup.cer
Select file ...
Back
Next

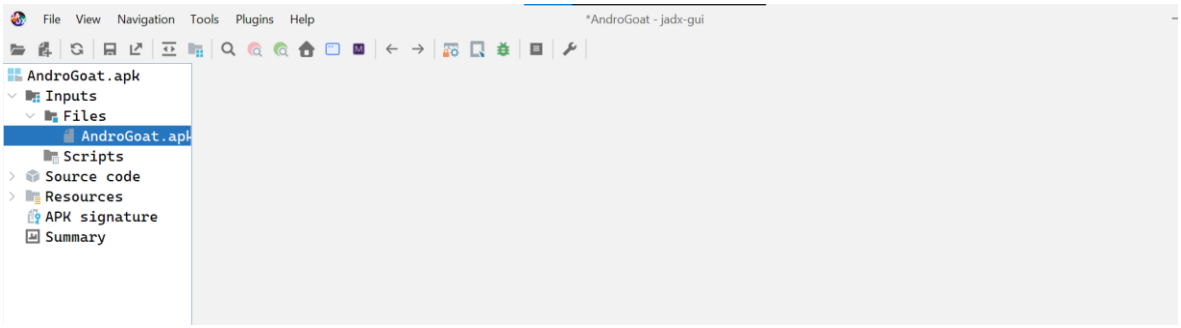




3. Analyse Statique

- Objectif : Comprendre la structure de l'application et identifier des points d'intérêt sans l'exécuter.
- Outil : JADX-GUI. Comment faire :

- Ouvrez l'APK d'AndroGoat avec JADX-GUI.



- Examinez le fichier AndroidManifest.xml pour comprendre les permissions demandées, les activités exportées, les services, les broadcast receivers et les content providers. Portez une attention particulière aux intent-filters pour les deeplinks.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    android:versionCode="1"
    android:versionName="1.0"
    package="owasp.sat.agoat">
    <uses-sdk
        android:minSdkVersion="18"
        android:targetSdkVersion="26"/>
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
    <application
        android:theme="@style/AppTheme"
        android:label="@string/app_name"
        android:icon="@mipmap/ic_launcher"
        android:debuggable="true"
        android:allowBackup="true"
        android:supportsRtl="true"
        android:networkSecurityConfig="@xml/network_security_config"
        android:roundIcon="@mipmap/ic_launcher_round">
        <activity android:name="owasp.sat.agoat.SplashActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
        <activity
            android:label="@string/app_name"
            android:name="owasp.sat.agoat.MainActivity"/>
        <activity
            android:label="@string/root"
            android:name="owasp.sat.agoat.RootDetectionActivity"/>
        <activity
            android:label="@string/logging"
            android:name="owasp.sat.agoat.InsecureLoggingActivity"/>
        <activity
```

comprendre les permissions demandées

```
package="owasp.sat.agoat">
<uses-sdk
    android:minSdkVersion="18"
    android:targetSdkVersion="26"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<application
```

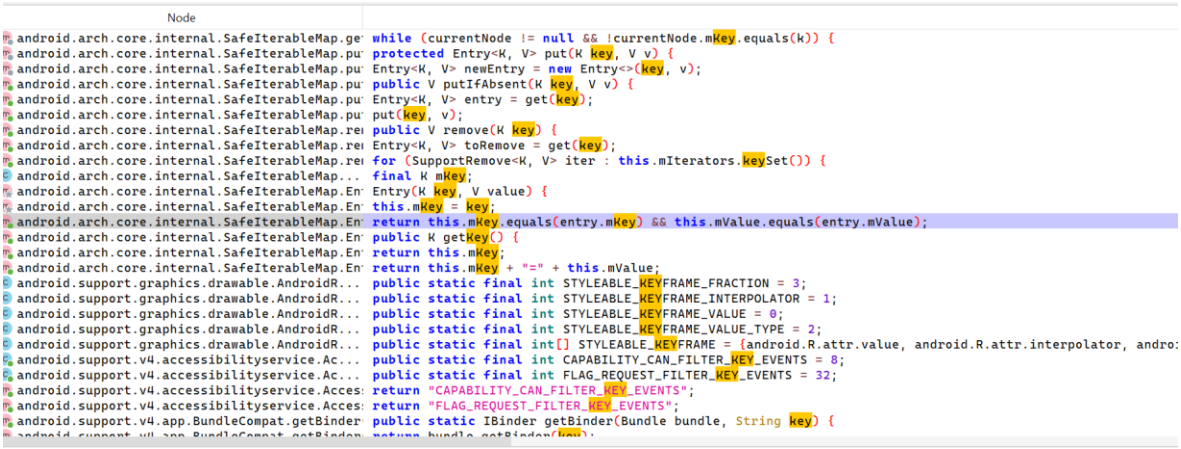
INTERNET

WRITE/READ_EXTERNAL_STORAGE

Services

```
<service
    android:name="owasp.sat.agoat.DownloadInvoiceService"
    android:enabled="true"
    android:exported="true"/>
```

- Naviguez dans le code source décompilé (Java). Recherchez des mots-clés suspects (password, key, token, secret, TODO, etc.).



Tutoriel à suivre : Pentesting Mobile : Guide pas à pas de l'évaluation AndroGoat

Liens : https://infosecwriteups.com/mobile-pentesting-androgoat-assessment-walkthrough-1a63a7edc677